

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบจัดการแพปลา

Fishery Management System



โดย

นาย กุลวงศ์ มากพานิชย์วัฒน์

นาย รัตน์ สุวรรค์มี

เลขหมู่.....  
เลขทะเบียน..... 62554  
วัน,เดือน,ปี..... 19 ส.ค. 2549

b. 1162124  
i. ....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**FISHERY MANAGEMENT SYSTEM**



**BY**  
**MR. KULLAWONG MAGPANITCHAWUT**  
**MR. RUT SURARUTSAMEE**

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF**  
**THE REQUIREMENT FOR THE DEGREE OF**  
**BACHEL IN DEPARTMENT OF INFORMATION ENGINEERING**  
**FACULTY OF ENGINEERING**  
**KING MONGKUT'S INSTITUTE OF TECHNOLOGY**  
**LADKRABANG**

**2005**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ใบเสนอปริญญาบัตร

หัวข้อปริญญาบัตร	ระบบจัดการแฟปลา
ชื่อนักศึกษา	1. นาย กุลวงศ์ มากพานิชย์วัฒน์ รหัสประจำตัว 46015618
	2. นาย รัตน์ สุวรรณี รหัสประจำตัว 46015636
อาจารย์ที่ปรึกษา	รศ. อรลภ แสงอรุณ อาจารย์ วันวิสา ชัชวงษ์
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2548

ปริญญาบัตรฉบับนี้ได้รับการอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบัง

(รองศาสตราจารย์ อรลภ แสงอรุณ)

อาจารย์ที่ปรึกษา

(อาจารย์ วันวิสา ชัชวงษ์)

อาจารย์ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบจัดการแฟปลา
ชื่อนักศึกษา	1. นาย กุลวงศ์ มากพานิชย์วัฒน์ รหัสประจำตัว 46015618 2. นาย รัตน์ สุวรรณี รหัสประจำตัว 46015636
อาจารย์ที่ปรึกษา	รศ. อรลภ แสงอรุณ อาจารย์ วันวิสา ชัชวงษ์
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2548

### บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการศึกษาระบบจัดการแฟปลา โดยได้มีการนำเอาข้อมูลของแฟมาทำการวิเคราะห์เพื่อออกแบบเป็นโปรแกรมโดยใช้ แผนภาพและกระแสข้อมูล (Data Flow Diagram , DFD) วิเคราะห์ทิศทางการไหลของข้อมูล ส่วนการออกแบบจำลองฐานข้อมูลใช้ E-R Model ภาษาที่ใช้ในการเขียนโปรแกรมใช้ Visual Basic.Net ส่วนในตัวโปรแกรมที่ได้ทำการออกแบบ มีส่วนของการเก็บข้อมูลของเจ้าของเรือ เก็บข้อมูลของลูกเรือ เก็บข้อมูลประเภทสินค้า เก็บข้อมูลของลูกค้าที่ต้องการสมัครเป็นสมาชิกในการใช้บริการ รวมถึงยังได้ออกแบบการใช้งานของส่วนการขายสินค้าให้กับลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**THESIS TITLE** FISHERY MANAGEMENT SYSTEM  
**STUDENT** MR. KUNLAWONG MAGPANITCHAWUT NO.46015618  
MR. RUT SURARUTSAMEE NO.46015636  
**ADVISOR** ASSOC.PROF. ORNLARP SANGALOON  
MISS. VANVISA CHUTCHAVONG  
**DEPARTMENT** INFORMATION ENGINEERING  
**YEAR** 2005

### Abstract

This project presents a study intensively Fishery Management System. By the data is study intensively writing programmed. It concern theory use Data Flow Diagram and E-R Model. Program develops by Visual Basic.Net. It keeps data such as fish, fishing boat, sell and buy. The fish keeps data name, price per kilogram, update quantity everyday. The fishing boat keeps data name, owner, date start, date get into.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี จากความช่วยเหลืออย่างยิ่งของ รองศาสตราจารย์ อรลภก แสงอรุณ อาจารย์ วันวิสา ชัชวงษ์ ซึ่งเป็นอาจารย์ที่ปรึกษาปริญญา นิพนธ์ ซึ่งท่านได้ให้คำแนะนำ ข้อคิดเห็น ต่างๆรวมถึงให้ข้อมูลในการทำปริญญาานิพนธ์ครั้งนี้ เป็นอย่างดี ขอขอบคุณอาจารย์ภาควิชาวิศวกรรมสารสนเทศที่ได้ให้ความรู้ตลอดหลักสูตรที่ได้ศึกษา จนกระทั่งสามารถนำความรู้ดังกล่าวมาประยุกต์ให้เกิดเป็นปริญญาานิพนธ์ นี้ขึ้นมาได้ ขอขอบคุณ เพื่อนๆ ที่คอยเป็นกำลังใจ รวมถึงให้คำแนะนำต่างๆ

สุดท้ายขอขอบคุณ บิดา มารดา ซึ่งเป็นบุคคล ที่ให้ทั้งกำลังใจที่หล่อเลี้ยง รวมถึง คำปรึกษาที่ดีๆ ให้สามารถผ่านพ้นบงเวลาที่ท้อใจ และให้การสนับสนุนทุกด้าน

คณะผู้จัดทำ

นาย กุลวงศ์ มากพานิชวัฒน์

นาย รัตน์ สุวรรค์มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศสารบัญ	ค
สารบัญ	ง
สารบัญรูป	ฉ
สารบัญตาราง	ช
บทที่ 1 บทนำ	1
1.1 แนวคิด และ ที่มาของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	1
1.4 ผลที่คาดว่าจะได้รับ	1
1.5 อุปกรณ์ที่ต้องใช้	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 แผนภาพและกระแสข้อมูล (Data Flow Diagram , DFD)	4
2.1.1 ส่วนประกอบของแผนภาพและกระแสข้อมูล	4
2.2 การออกแบบฐานข้อมูลโดยใช้แบบจำลอง E-R Model	6
2.2.1 เอนทิตี (Antity)	6
2.2.2 เอนทิตีชนิดอ่อน (Weak Entity)	7
2.2.3 แอททริบิวต์ (Attribute)	7
2.2.4 แอททริบิวต์ผสม ( Composite Attribute )	7
2.2.5 แอททริบิวต์ที่ถูกแปลงค่ามา ( Derived Attribute )	8
2.2.6 มัลติแวลูแอททริบิวต์ ( Multi - valued Attribute )	8
2.2.7 ความสัมพันธ์ ( Relationship )	8
2.3 ความสัมพันธ์ระหว่างเอนทิตี	9
2.3.1 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง ( One - to - One Relationship )	9
2.3.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม ( One - to - Many Relationship )	10
2.3.3 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม ( Many - to - Many Relationship )	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 Binary Relationship	11
2.4.1 N - Ary Relationship	11
2.4.2 Recursive Relationship	11
2.4.3 คาคินาลิตีของความสัมพันธ์ ( Cardinality of Relationships )	11
2.5 รูปแบบของฐานข้อมูล	13
2.5.1 ฐานข้อมูลเชิงสัมพันธ์ ( Relational Database )	13
2.5.2 ฐานข้อมูลแบบลำดับชั้น ( Hierarchical Relational )	13
2.5.3 ฐานข้อมูลแบบเครือข่าย ( Network Relational )	13
2.6 ภาษาและโปรแกรมที่ใช้ในการพัฒนา	14
2.6.1 ในVB.NETมีชนิดของโปรเจกต์ดังต่อไปนี้	16
2.7 OOP (Object Oriented Programming)	17
2.7.1 Class และ Object	17
2.7.2 Member Class	18
2.7.3 Constructor และ Destructor	18
2.7.4 คุณสมบัตินี้ของ OOP	19
2.8 ฐานข้อมูลที่ใช้ ระบบฐานข้อมูลเชิงสัมพันธ์ (SQL Server 2000)	19
2.8.1 ส่วนประกอบของฐานข้อมูล	20
2.8.2 Microsoft SQL Sever	20
2.8.3 ส่วนประกอบพื้นฐานของฐานข้อมูลเชิงสัมพันธ์	21
บทที่3 การออกแบบระบบ	23
3.1 วิเคราะห์ระบบ	23
3.2 แผนภาพและกระแสข้อมูล (Data Flow Diagram , DFD)	24
3.3 E-R Model ของระบบฐานข้อมูล	28
3.4 คาค้าดิกชันนารี (Data Dictionary)	29
บทที่4 การทดลองและผลการทดลอง	34
บทที่5 สรุปผลการทดลอง	50
บรรณานุกรม	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- ชัชวาล ศุภเกษม. คัมภีร์การใช้ Visual Basic . Net ฉบับสมบูรณ์. กรุงเทพฯ. ซีเอ็ดบุ๊คเซ็น,2545  
 นันทนี แวงโสภา. อินไซด์ Visual Basic และ VB.NET ฉบับ Database. กรุงเทพฯ. โปรวิชั่น, 2544  
 กิตติ ภักดีวัฒนา. คัมภีร์ระบบฐานข้อมูล. กรุงเทพฯ. ดวงกมลสมัย,2542  
 ผศ. สมจิตร อาจอินทร์. ระบบฐานข้อมูล. ขอนแก่น.ศูนย์หนังสือมหาวิทยาลัยขอนแก่น,2541



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่	หน้า
รูปที่ 2.1 แผนภาพกระแสข้อมูลต่างระดับ	3
รูปที่ 2.2 สัญลักษณ์ ของกระบวนการการ (Process)	4
รูปที่ 2.3 สัญลักษณ์กระแสข้อมูล(Data Flow)	5
รูปที่ 2.4 สัญลักษณ์ของแหล่งเก็บข้อมูล(Data Store)	5
รูปที่ 2.5 สัญลักษณ์ของสิ่งที่อยู่นอกระบบ(External Entity)	5
รูปที่ 2.6 เอนตีตี้	6
รูปที่ 2.7 เอนตีตี้ชนิดอ่อนแอ	7
รูปที่ 2.8 แอททริบิวท์	7
รูปที่ 2.9 แอททริบิวท์ผสม	8
รูปที่ 2.10 แอททริบิวท์ที่ถูกแปลงค่ามา	8
รูปที่ 2.11 มัลติแวลูตแอททริบิวท์	8
รูปที่ 2.12 ความสัมพันธ์	8
รูปที่ 2.13 Total Participation	9
รูปที่ 2.14 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง	9
รูปที่ 2.15 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม	10
รูปที่ 2.16 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม	10
รูปที่ 2.17 ความสัมพันธ์ให้อยู่ในรูปหนึ่งต่อกลุ่ม	10
รูปที่ 2.18 Binary Relationship	11
รูปที่ 2.19 N - Ary Relationship	11
รูปที่ 2.20 Recursive Relationship	11
รูปที่ 2.21 คาดินาลิตี้ของความสัมพันธ์ (1,N)	12
รูปที่ 2.22 คาดินาลิตี้ของความสัมพันธ์ (M,N)	12
รูปที่ 3.1 Data Flow Diagram Level 0	24
รูปที่ 3.2 Data Flow Diagram Level 1	25
รูปที่ 3.3 Data Flow Diagram Level 1 ส่วนที่ 1	25
รูปที่ 3.4 Data Flow Diagram Level 1 ส่วนที่ 2	26
รูปที่ 3.5 Data Flow Diagram Level 1 ส่วนที่ 3	26
รูปที่ 3.6 Data Flow Diagram Level 1 ส่วนที่ 4	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่	หน้า
รูปที่ 4.1 แสดงหน้าต่าง Login เข้าสู่ระบบ	34
รูปที่ 4.2 หน้าหลักของตัวโปรแกรม	35
รูป 4.3 แสดงจุดประสงค์ในการทำโปรแกรมระบบจัดการแพปลา	35
รูปที่ 4.4 แสดงข้อมูลและรายละเอียดของสัตว์น้ำแต่ละชนิด	36
รูปที่ 4.5 ส่วนของการกำหนดราคาซื้อจากเรือและราคาที่ขายให้ลูกค้า	37
รูปที่ 4.6 แสดงส่วนของการเพิ่มชนิดข้อมูลของสัตว์น้ำ	38
รูปที่ 4.7 แสดงการเก็บข้อมูลรายละเอียดของเรือ	39
รูปที่ 4.8 แสดงการเก็บข้อมูลรายละเอียดของลูกเรือ	40
รูปที่ 4.9 แสดงส่วนการเก็บข้อมูลของสมาชิก	41
รูปที่ 4.10 แสดงหน้าต่างของการต่ออายุความเป็นสมาชิก	42
รูปที่ 4.11 แสดงหน้าต่างของการปรับปรุงแก้ไขสิทธิประโยชน์ของสมาชิก	43
รูปที่ 4.12 แสดงหน้าต่างของการขายสัตว์น้ำให้ลูกค้า	44
รูปที่ 4.13 แสดงใบเสร็จออกให้กับลูกค้า	45
รูปที่ 4.14 แสดงหน้าต่างของการรับข้อมูลของสัตว์ จากเรือประมง	46
รูปที่ 4.15 แสดงข้อมูลใบเสร็จที่ออกให้กับเรือประมง	47
รูปที่ 4.16 แสดงการคิดคำนวณรายได้ให้กับลูกเรือ	48
รูปที่ 4.17 แสดงการตรวจสอบความเป็นสมาชิก	49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่	หน้า
ตาราง 3.1 สำหรับเก็บสาขา	29
ตาราง 3.2 สำหรับเก็บประเภทสินค้า	29
ตาราง 3.3 สำหรับเก็บจำนวนสินค้าทั้งหมด	29
ตาราง 3.4 สำหรับเก็บรายละเอียดของสินค้า	29
ตาราง 3.5 สำหรับเก็บสีของเรือ	30
ตาราง 3.6 สำหรับเก็บรายชื่อสมาชิก	30
ตาราง 3.7 สำหรับเก็บรายละเอียดของสินค้า	30
ตาราง 3.8 สำหรับเก็บประเภทสมาชิก	31
ตาราง 3.9 สำหรับเก็บสินค้าที่ซื้อจากรีอ	31
ตาราง 3.10 สำหรับเก็บรายละเอียดการซื้อ	31
ตาราง 3.11 สำหรับเก็บตำแหน่งลูกเรือ	32
ตาราง 3.12 สำหรับเก็บการขายสินค้า	32
ตาราง 3.13 สำหรับเก็บรายละเอียดการขาย	32
ตาราง 3.14 สำหรับเก็บขนาดของเรือ	32
ตาราง 3.15 สำหรับเก็บข้อมูลเจ้าของเรือ	33
ตาราง 3.16 สำหรับเก็บข้อมูลลูกเรือ	33
ตาราง 3.17 สำหรับเก็บLogin	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 1

### บทนำ

#### 1.1 แนวคิด และ ที่มาของปัญหา

ระบบการจัดการแพปลา สามารถแบ่งฐานข้อมูลออกเป็น 2 ส่วน คือ ส่วนของเรือประมง และ ส่วนของการซื้อขายปลา ในส่วนของเรือประมงประกอบด้วย ฐานข้อมูลของจำนวนเรือ ชื่อเรือ จำนวนลูกเรือ ชื่อลูกเรือ รายรับรายจ่ายของเรือแต่ละลำ และ การแบ่งอัตราส่วนจากรายได้ของลูกเรือแต่ละคน ส่วนของระบบการซื้อขายปลา ประกอบด้วยฐานข้อมูลชนิด ของปลาขนาดของปลา ซึ่งราคาขายจะแบ่งตาม ชนิด และขนาดของปลา นอกจากนี้ยังมีองค์ประกอบของความสดของปลาเป็นตัวแปรที่สำคัญ และระบบการจัดการฐานข้อมูล

#### 1.2 วัตถุประสงค์

1.2.1 เพื่อศึกษาระบบพาณิชย์อิเล็กทรอนิกส์

1.2.2 เพื่อศึกษาระบบการจัดการฐานข้อมูล

1.2.4 เพื่อเรียนรู้การพัฒนาภาษาในการเขียนโปรแกรมโดย ภาษา Visual Basic.Net และใช้ SQL Server 2000 เป็นฐานข้อมูล

#### 1.3 ขอบเขตของโครงการ

1.3.1 ส่วนของเรือประมง มีการเก็บข้อมูล ชื่อเรือ ชื่อลูกเรือ ชื่อเจ้าของเรือ สีเรือ วันที่เรือออกทำ วันที่เรือเข้าท่า จำนวนของปลาที่ทำได้แต่ละครั้ง

1.3.2 ส่วนของการเก็บข้อมูล ปลา คือเก็บชื่อปลา ราคาปลาที่รับซื้อมาจากเรือประมงต่อ กิโลกรัม ราคาปลาที่ขายให้ลูกค้าต่อกิโลกรัม จำนวนของปลาที่เหลือในแพปลา

1.3.3 การเก็บข้อมูล สมาชิก ที่มาซื้อปลาโดยจะได้สิทธิพิเศษจากการสมัคร โดยการได้รับส่วนลด มีสมาชิกสองแบบคือ แบบธรรมดา แบบVIP

#### 1.4 ผลที่คาดว่าจะได้รับ

1.4.1 ได้รับความรู้ด้านการเขียนโปรแกรมแอปพลิเคชัน รวมทั้งการจัดการระบบฐานข้อมูล

1.4.2 ระบบที่พัฒนาสามารถนำไปประยุกต์ใช้งานได้จริง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับงานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5 อุปกรณ์ที่ต้องใช้

### 1.5.1 ฮาร์ดแวร์

- เครื่องคอมพิวเตอร์ สำหรับพัฒนา โปรแกรม จำนวน 1 เครื่อง

### 1.5.2 ซอฟต์แวร์

- Visual Basic.Net เป็นภาษาที่ใช้ในการพัฒนา โปรแกรม

- SQL Sever 2000 เป็นภาษาที่ใช้ในการจัดการฐานข้อมูล



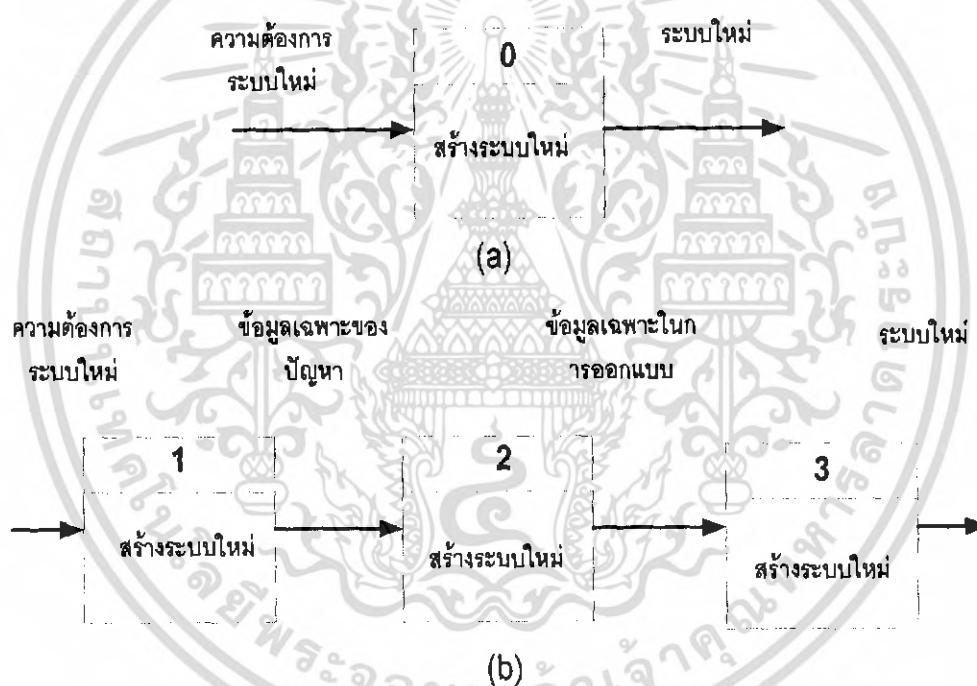
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 แผนภาพและกระแสข้อมูล (Data Flow Diagram , DFD)

แผนภาพกระแสข้อมูล เป็นเครื่องมือใช้กันอย่างแพร่หลายในการเขียนระบบใหม่ โดยเฉพาะกับระบบที่มี “กระบวนการ” ของระบบที่มีความสำคัญและมีความสลับซับซ้อนมากกว่าข้อมูลที่ไหลเข้า



รูปที่ 2.1 แผนภาพกระแสข้อมูลต่างระดับ

จากรูปที่ 2.1 ลูกศรแสดงถึงข้อมูลซึ่งจะต้องมีชื่อกำกับลูกศรนี้จะเคลื่อนผ่านระบบตรงกลางแสดงถึงการกระทำสิ่งใดสิ่งหนึ่งกับข้อมูลที่วิ่งเข้ามาและลูกศรที่วิ่งออกจากระบบแสดงถึงผลลัพธ์ที่ได้จากการกระทำบนข้อมูลนั้นๆ แต่จากข้อมูลนี้เราไม่อาจทราบได้ว่าระบบใหม่สร้างขึ้นมาได้อย่างไรเพื่อให้ได้รายละเอียดมากขึ้น ในการสร้างระบบใหม่เราจะทำการแตกระบบนี้ลงอีกระดับหนึ่งในรูป 2.1 (b) ซึ่งจะแตกออกเป็นอีก3กระบวนการ (Process) หมายถึงมีการกระทำอยู่ 3 ขั้นตอนในการสร้างระบบใหม่ให้ได้รายละเอียดมากขึ้น ทำให้เข้าใจระบบนั้นได้ดีขึ้น และอาจจะแตกกระบวนการทั้งสามลงอีกเพื่อให้มีรายละเอียดมากขึ้น

เอกสารนี้เป็นทรัพย์สินทางปัญญาของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย

ขึ้นอีกก็ได้จนกว่ารายละเอียดที่เหมาะสมตามจุดประสงค์ของเรา วิธีการเขียนนี้ก็เหมือนกับการเขียนโปรแกรมคอมพิวเตอร์ ซึ่งมีหลายโปรแกรมย่อยนั่นเอง ซึ่งจะช่วยให้การเขียนโปรแกรมใหญ่ๆง่ายขึ้น ดังนั้นการเขียนแผนภาพและกระแสข้อมูลละเอียดเท่าใด ก็ทำให้การออกแบบที่ตามมาง่ายยิ่งขึ้นเท่านั้น

### 2.1.1 ส่วนประกอบของแผนภาพและกระแสข้อมูล

กระบวนการ (Process) การประมวลผล คือ งานที่ต้องทำ แทนด้วยสัญลักษณ์ดังรูป



รูปที่ 2.2 สัญลักษณ์ ของกระบวนการการ (Process)

กระบวนการเปลี่ยนข้อมูลขาเข้าเป็นผลลัพธ์ นั้นหมายความว่า จะต้องมีการกระทำบางอย่างต่อข้อมูล ทำให้เกิด ผลลัพธ์ขึ้นมา โดยปกติแล้วข้อมูลที่เข้าสู่กระบวนการ จะแตกต่างจากข้อมูลเมื่อออกจากกระบวนการ

กระบวนการเป็นตัวอย่างหนึ่งของ “กล่อง” หมายถึง เราทราบว่าข้อมูลเป็นอะไร ผลลัพธ์ที่เราต้องการ และหน้าที่ โดยทั่วไปของกระบวนการ แต่จะไม่ทราบว่า กระบวนการนั้นทำงานอย่างไร

หลักการของกล่องคำมีประโยชน์ในการเขียนแผนภาพ แสดงการเปลี่ยนแปลงแผนภาพแสดงการเปลี่ยนแปลงข้อมูล โดยที่ยังไม่ต้องทราบรายละเอียดว่า กระบวนการ นั้นมีรายละเอียดอะไรบ้าง ซึ่งหารายละเอียดเหล่านี้ได้ภายหลังชื่อกระบวนการเป็นตัวบอกว่า กระบวนการนั้นทำหน้าที่อะไร คำที่ใช้ควรจะมี ความหมายที่แน่นอนและควรใช้คำกริยา เช่น คำนวณ แก้ไข บันทึก ยืนยันข้อมูล เป็นต้น ถ้าการทำงานใดที่เราไม่สามารถหาคำแทนได้อย่างเหมาะสมอาจหมายความว่างานนั้นไม่ใช่กระบวนการก็ได้

กระแสข้อมูล (Data Flow) กระแสข้อมูลแทนด้วยสัญลักษณ์ศร โดยมีชื่อข้อมูลการ กับบนหัวลูกศร ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### รูปที่ 2.3 สัญลักษณ์กระแสข้อมูล(Data Flow)

ข้อมูลจะไหลระหว่างกระบวนการต่างๆ และอาจ จะเคลื่อนที่ออกมาจากสิ่งที่อยู่นอกระบบก็ได้ข้อมูลที่เคลื่อนที่อาจจะเป็นข้อมูลเดี่ยวๆ เช่น เลขที่สินค้า หรือกลุ่มของข้อมูล เช่น ข้อมูลพนักงาน ข้อมูลลูกค้า อาจมีรายละเอียดเป็นชื่อลูกค้า เลขที่ ที่อยู่ และหากต้องการอ้างอิงข้อมูลสองส่วน ที่ไม่เกี่ยวข้องกันให้เขียนเป็นลูกศร 2 อัน ข้อมูลแต่ละอันหรือกลุ่มข้อมูลควรมีชื่อของตัวเองที่ไม่เหมือนกันควรหลีกเลี่ยงการใช้ชื่อที่กว้างเกินไป เช่น “ ข้อมูลผิดพลาด” เพราะว่า ในระบบหนึ่งๆอาจจะมีชื่อผิดพลาดได้หลายแห่งเราควรใช้ชื่อที่เฉพาะเจาะจงมากกว่านี้ให้ชัดเจน

แหล่งเก็บข้อมูล(Data Store) แหล่งเก็บข้อมูลแสดงด้วยสัญลักษณ์ ดังรูป



### รูปที่ 2.4 สัญลักษณ์ของแหล่งเก็บข้อมูล(Data Store)

ข้อมูลที่เก็บไว้ จะถูกเรียกใช้เมื่อต้องการ ถ้าหัวลูกศรวิ่งเข้าสู่ แหล่งเก็บข้อมูล แสดงว่ามีการเขียนข้อมูล ถ้าลูกศรวิ่งออกจากแหล่งเก็บข้อมูล แสดงว่ามีการอ่านข้อมูล การตั้งแหล่งเก็บข้อมูลควรเป็นคำนาม

สิ่งที่อยู่นอกระบบ(External Entity)สิ่งที่อยู่นอกระบบแสดงด้วยรูปสี่เหลี่ยมผืนผ้าดังรูป



### รูปที่ 2.5 สัญลักษณ์ของสิ่งที่อยู่นอกระบบ(External Entity)

สิ่งที่อยู่นอกระบบ ส่วนใหญ่จะเป็นตัวบุคคลหรือองค์กรต่างๆสิ่งที่อยู่นอกระบบอาจจะเป็นที่ส่งข้อมูลเข้าระบบ หรืออาจเป็นที่รับข้อมูลจากระบบก็ได้เราไม่สนใจการทำงานของสิ่งที่อยู่นอกระบบถึงแม้ว่าการติดต่อผ่านทางข้อมูล เราจะสนใจเฉพาะข้อมูลที่เข้าสู่ระบบหรือออกเอกสารนี้เป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากระบบสู่ภายนอกเท่านั้น สัญลักษณ์ต่างๆข้างต้น จะถูกนำมาเขียนเป็น Data Flow Diagram ของระบบโดยแบ่งออกเป็นระดับ (Level) ดังนี้

คอนเท็กซ์ไดอะแกรม(Context Diagram) คอนเท็กซ์ไดอะแกรมเป็นแผนภาพกระแสข้อมูล ระดับสูงสุดของระบบ ซึ่งในระดับนี้จะบอกว่าระบบที่เราสนใจมีอินพุตเป็นอะไร ผลลัพธ์ที่ได้จากระบบคืออะไร และข้อมูลที่เกี่ยวข้องกับระบบภายนอก โดยจะยังไม่แสดงแหล่งเก็บข้อมูล ออกมา ซึ่งแหล่งเก็บข้อมูลจะถูกแสดงในลำดับถัดไป

แผนภาพข้อมูลในลำดับรอง(Lower Level Data Flow Diagram)แผนภาพกระแสข้อมูล จะมีการเขียนแตกออกเป็นลำดับ โดยเริ่มจากระดับ 0 ซึ่งเป็นการเขียนภาพรวมทั้งหมดของระบบ จากนั้นแตกรายละเอียดลงไปในแต่ละส่วน เป็นแผนภาพกระแสข้อมูลในระดับรองๆ ลงไปเพื่อให้ได้รายละเอียดในแต่ละส่วนมากขึ้น

## 2.2 การออกแบบฐานข้อมูลโดยใช้แบบจำลอง E-R Model

ในการออกแบบฐานข้อมูลขึ้นใช้ในระบบงานสารสนเทศใดๆ จะต้องอาศัย Data Model เพื่อนำเสนอรายละเอียดต่างๆที่เกี่ยวข้องกับข้อมูลในฐานข้อมูลที่ออกแบบ เนื่องจาก Data Model เป็นแบบจำลองที่มีรูปแบบในการนำเสนอรายละเอียดต่างๆ ที่เกี่ยวข้องกับฐานข้อมูลที่เป็นมาตรฐาน จึงทำให้สามารถนำเสนอต่อผู้ใช้ในแต่ละระดับที่มีมุมมองที่แตกต่างกันได้เป็นอย่างดี สำหรับ Data Model ที่นิยมใช้ได้แก่ Entity- Relationship Model หรือเรียกสั้นๆว่า E- R Model โดยจะมีการนำเสนอโครงสร้างของฐานข้อมูลในระดับแนวคิด (Conceptual Level) ออกมา ในลักษณะของแผนภาพ (Diagram) ที่มีโครงสร้างที่ง่ายต่อการทำความเข้าใจ ทำให้สามารถมองเห็นภาพรวมของเอนทิตีทั้งหมดที่มีในระบบฐานข้อมูล รวมทั้งความสัมพันธ์ระหว่างเอนทิตีเหล่านั้น และนอกจากนี้ยังเป็นแผนภาพที่ไม่อิงกับระบบจัดการฐานข้อมูลที่มีโมเดลฐานข้อมูลแบบใดไม่ว่าจะเป็น Relationship , Hierarchical หรือ Network Model

### 2.2.1 เอนทิตี (Antity)

คือชื่อของสิ่งใดสิ่งหนึ่งอาจเกี่ยวกับคนสถานที่สิ่งของการกระทำซึ่งต้องการ จัดเก็บข้อมูลไว้เช่นเอนทิตีของพนักงานสินค้า ลูกค้า การสั่งซื้อ เป็นต้น เอนทิตีจะใช้สัญลักษณ์สี่เหลี่ยมผืนผ้า โดยใส่ชื่อของเอนทิตีซึ่งจะใช้คำนามไว้ภายใน

พนักงาน

รูปที่ 2.6 เอนทิตี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.2 เอนทิตีชนิดอ่อน (Weak Entity)

เอนทิตีนี้มีลักษณะต่างจากเอนทิตีทั่วไปคือสมาชิกของเอนทิตีนี้จะสามารถมีคุณสมบัติที่บ่งบอกเอกลักษณ์ของแต่ละสมาชิกได้นั้น จะต้องอาศัยคุณสมบัติใดคุณสมบัติหนึ่งของเอนทิตีทั่วไปมาประกอบกับคุณสมบัติของตัวเองเอนทิตีนี้จะไม่มีความหมาย หากไม่มีเอนทิตีอื่นในฐานะข้อมูล (Parent Entity) เช่น เอนทิตีประวัติครอบครัวของพนักงานเป็นต้น เอนทิตีอ่อนจะใช้สัญลักษณ์

กรอบคว่ำ

### รูปที่ 2.7 เอนทิตีชนิดอ่อนแอ

การกำหนดแอททริบิวต์ที่เป็นคีย์หลักของเอนทิตีอ่อนแอจำเป็นต้องประกอบด้วยแอททริบิวต์บางตัวจากเอนทิตีอื่น (parent entity) กับแอททริบิวต์ที่มีในเอนทิตีของตัวเอง

### 2.2.3 แอททริบิวต์ (Attribute)

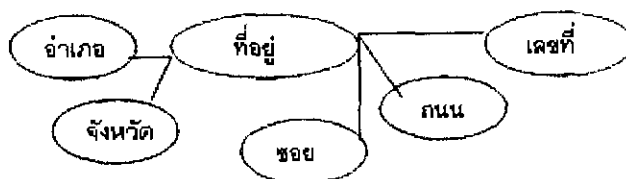
คือรายละเอียดของเอนทิตีหนึ่ง เช่น เอนทิตีพนักงานประกอบด้วยแอททริบิวต์รหัสพนักงานชื่อ เงินเดือน เป็นต้น แอททริบิวต์จะใช้สัญลักษณ์

ที่อยู่

### รูปที่ 2.8 แอททริบิวต์

### 2.2.4 แอททริบิวต์ผสม (Composite Attribute)

คือแอททริบิวต์ที่ประกอบด้วยข้อมูลหลายส่วนมารวมกันหรืออาจแยกเป็น ชื่อแอททริบิวต์ย่อยได้อีก เช่น แอททริบิวต์ที่อยู่สามารถแยกได้เป็น เลขที่ ถนน ซอย อำเภอ จังหวัด และรหัสไปรษณีย์ เป็นต้น



หรือ

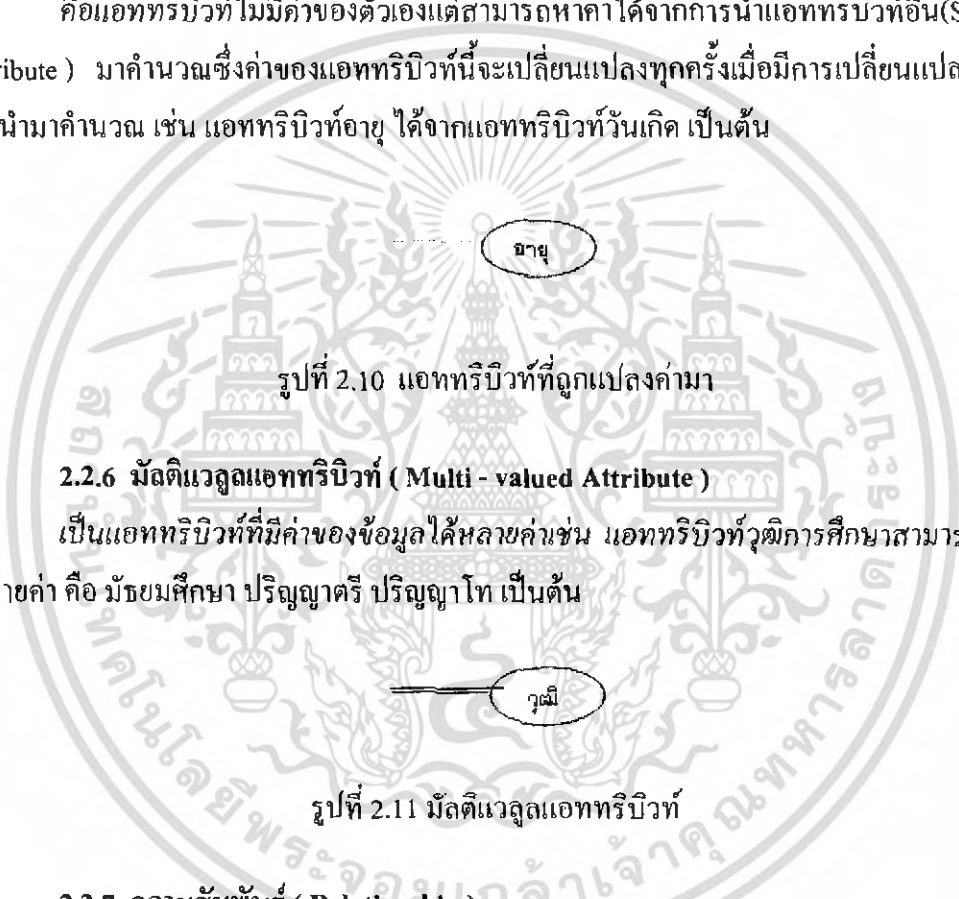
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 แอททริบิวต์ผสม

### 2.2.5 แอททริบิวต์ที่ถูกแปลงค่ามา ( Derived Attribute )

คือแอททริบิวต์ที่ไม่มีค่าของตัวเองแต่สามารถหาค่าได้จากการนำแอททริบิวต์อื่น (Stored attribute) มาคำนวณซึ่งค่าของแอททริบิวต์นี้จะเปลี่ยนแปลงทุกครั้งเมื่อมีการเปลี่ยนแปลงค่าที่ถูกนำมาคำนวณ เช่น แอททริบิวต์อายุ ได้จากแอททริบิวต์วันเกิด เป็นต้น



อายุ

รูปที่ 2.10 แอททริบิวต์ที่ถูกแปลงค่ามา

### 2.2.6 มัลติแวลูแอททริบิวต์ ( Multi - valued Attribute )

เป็นแอททริบิวต์ที่มีค่าของข้อมูลได้หลายค่าเช่น แอททริบิวต์วุฒិการศึกษาสามารถมีค่าหลายค่า คือ มัธยมศึกษา ปริญญาตรี ปริญญาโท เป็นต้น

วุฒิ

รูปที่ 2.11 มัลติแวลูแอททริบิวต์

### 2.2.7 ความสัมพันธ์ ( Relationship )

คือ คำกริยาที่แสดงความสัมพันธ์ระหว่างสองเอนติตี้ เช่น เอนติตี้พนักงาน และเอนติตี้แผนก มีความสัมพันธ์กันในด้าน "สังกัดอยู่" คือ พนักงานแต่ละคนทำงานสังกัดอยู่ในแผนกใดแผนกหนึ่ง เป็นต้น ความสัมพันธ์จะใช้สัญลักษณ์



รูปที่ 2.12 ความสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอนทิตีที่เกี่ยวข้องกับความสัมพันธ์เรียกว่า Participant ของความสัมพันธ์ และจำนวนของ Participant ในความสัมพันธ์เรียกว่า Degree ของความสัมพันธ์ (มีความหมายแตกต่างไปจาก Degree ในแบบจำลองข้อมูลเชิงสัมพันธ์) จำแนกได้ 2 ลักษณะ

Total Participation เป็นความสัมพันธ์ที่ทุกสมาชิกในเอนทิตีหนึ่งจะมีข้อมูลสัมพันธ์กับข้อมูลในอีกหนึ่งเอนทิตี เช่น อาจารย์ทุกคนต้องสังกัดคณะใดคณะหนึ่งเท่านั้น การระบุความสัมพันธ์แบบ Total Participation ใช้สัญลักษณ์เส้นคู่ เพื่อเชื่อมต่อระหว่างความสัมพันธ์กับเอนทิตีที่ทุกสมาชิกมีความสัมพันธ์กับอีกเอนทิตีหนึ่ง



รูปที่ 2.13 Total Participation

Partial Participation เป็นความสัมพันธ์ที่บางสมาชิกในเอนทิตีหนึ่งจะมีสัมพันธ์กับข้อมูลในอีกหนึ่งเอนทิตี เช่น มีนักศึกษาเพียงบางคนเท่านั้นที่ผู้แทนนักศึกษาในแต่ละคณะ ใช้สัญลักษณ์เช่นเดียวกับ การแสดงความสัมพันธ์ระหว่างเอนทิตีใดๆที่เกี่ยวข้อง

## 2.3 ความสัมพันธ์ระหว่างเอนทิตี

ความสัมพันธ์ระหว่างสองเอนทิตี แบ่งเป็น 3 ประเภท คือ

### 2.3.1 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One - to - One Relationship)

เป็นการแสดงความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่งว่า มีความสัมพันธ์กับข้อมูลอย่างมากหนึ่งข้อมูลกับอีกเอนทิตีหนึ่งในลักษณะที่เป็นหนึ่งต่อหนึ่งเช่น เอนทิตีนักศึกษากับเอนทิตีโครงการวิจัยมีความสัมพันธ์กันแบบหนึ่งต่อหนึ่งคือนักศึกษาแต่ละคนทำโครงการวิจัยได้ 1 โครงการเท่านั้นและแต่ละโครงการวิจัยมีนักศึกษารับผิดชอบได้ไม่เกิน 1 คน เป็นต้น



รูปที่ 2.14 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง

### 2.3.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม ( One - to - Many Relationship )

เป็นการแสดงความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่งว่ามีความสัมพันธ์กับข้อมูลหลายข้อมูลกับอีกเอนทิตีหนึ่ง เช่น ความสัมพันธ์ของลูกค้าและคำสั่งซื้อเป็นแบบหนึ่งต่อกลุ่ม คือลูกค้าแต่ละคนสามารถสั่งซื้อได้หลายคำสั่งซื้อ แต่แต่ละคำสั่งซื้อมาจากลูกค้าเพียงคนเดียว เป็นต้น



รูปที่ 2.15 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม

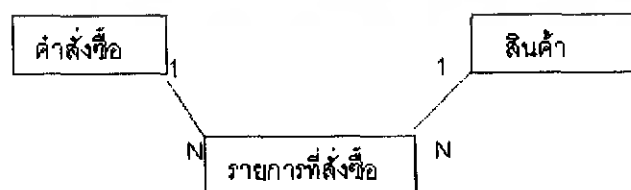
### 2.3.3 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม ( Many - to - Many Relationship )

เป็นการแสดงความสัมพันธ์ของข้อมูลของสองเอนทิตีในลักษณะแบบกลุ่มต่อกลุ่ม เช่น ความสัมพันธ์ระหว่างคำสั่งซื้อกับสินค้าเป็นแบบกลุ่มต่อกลุ่มคือแต่ละคำสั่งซื้ออาจสั่งซื้อสินค้าได้มากกว่าชนิดและในสินค้าแต่ละชนิดอาจปรากฏอยู่ในคำสั่งซื้อได้มากกว่าคำสั่งซื้อ



รูปที่ 2.16 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

ความสัมพันธ์แบบกลุ่มต่อกลุ่มเป็นเรื่องยุ่งยากในการออกแบบฐานข้อมูล โดยทั่วไปจะสร้างเอนทิตีใหม่ขึ้นมา เรียกว่า Gerund ( Composite Entity หรือ Intersection Entity หรือ Synthetic Entity ) เพื่อเชื่อมความสัมพันธ์ของสองเอนทิตีเดิม โดยมีวัตถุประสงค์เพื่อปรับความสัมพันธ์ให้อยู่ในรูปหนึ่งต่อกลุ่ม



รูปที่ 2.17 ความสัมพันธ์ให้อยู่ในรูปหนึ่งต่อกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้เราสามารถจัดประเภทของความสัมพันธ์ระหว่างเอนทิตีตามจำนวนของเอนทิตีได้ดังนี้

## 2.4 Binary Relationship

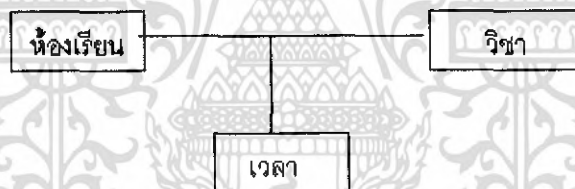
เป็นความสัมพันธ์ที่พบมากที่สุดโดยเป็นความสัมพันธ์ที่เกิดขึ้นระหว่าง 2 เอนทิตีใดๆ เช่น



รูปที่ 2.18 Binary Relationship

### 2.4.1 N - Ary Relationship

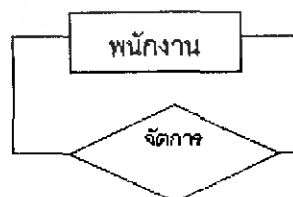
เป็นความสัมพันธ์ที่เกิดขึ้นระหว่าง เอนทิตีมากกว่า 2 เอนทิตีขึ้นไป เช่น



รูปที่ 2.19 N - Ary Relationship

### 2.4.2 Recursive Relationship

เป็นความสัมพันธ์ที่เกิดขึ้นกับเอนทิตีเดียว ในกรณีที่แอททริบิวต์ของเอนทิตีนั้นสามารถสร้างความสัมพันธ์กับอีกแอททริบิวต์หนึ่งภายในเอนทิตีเดียวกัน เช่น



รูปที่ 2.20 Recursive Relationship

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.3 คาคินาลิตีของความสัมพันธ์ ( Cardinality of Relationships )

ในการแสดงความสัมพันธ์ระหว่างเอนทิตีสามารถระบุจำนวนของเอนทิตีแต่ละประเภท ในความสัมพันธ์หนึ่งๆ โดยระบุเป็นค่าต่ำสุด และค่าสูงสุดของเอนทิตีแต่ละประเภท ซึ่งแบ่ง ออกได้ 4 ลักษณะโดยใช้สัญลักษณ์ดังนี้ที่ ค่าต่ำสุดและสูงสุดเป็น 1 คือบังคับให้มีค่าเป็น 1 ใช้ สัญลักษณ์(1,1) ท ค่าต่ำสุดเป็น 0 และสูงสุดเป็น 1 คืออาจจะไม่มีค่าหรือมีค่าเป็น 1 ใช้ สัญลักษณ์(0,1) ท ค่าต่ำสุดเป็น 0 และสูงสุดเป็นบวก คืออาจจะไม่มีค่า หรือมีค่าเท่าใดก็ได้ ใช้ สัญลักษณ์(0,M)เช่น(0,25) ท ค่าต่ำสุดเป็น 1 และสูงสุดเป็นบวก คือต้องมีค่าอย่างน้อยเป็น 1 ใช้ สัญลักษณ์(1,M)เช่น(1,32) ตัวอย่าง



รูปที่ 2.21 คาคินาลิตีของความสัมพันธ์ (1,N)

จากแผนภาพหมายถึง

อาจารย์หนึ่งท่านอาจจะไม่สอนวิชาใดๆเลย และจะสอนได้ไม่เกิน 2 วิชา ( คาคินาลิตี ของอาจารย์ คือ (0,2) )

แต่ละวิชาจะต้องมีอาจารย์ผู้สอน 1 ท่านต่อ 1 วิชาเท่านั้น ( คาคินาลิตีของวิชา คือ (1,1,)) ตัวอย่าง



รูปที่ 2.22 คาคินาลิตีของความสัมพันธ์ (M,N)

จากแผนภาพหมายถึง

ในแต่ละวิชาอาจจะไม่มีนักเรียนลงทะเบียนเรียนเลย และจะลงทะเบียนเรียนได้ไม่เกิน 60 คน ( คาคินาลิตีของวิชา คือ (0,60) )

นักเรียนแต่ละคนจะต้องลงทะเบียนเรียนอย่างน้อย 1 วิชา และจะลงทะเบียนเรียนอย่าง มากสุด 7 วิชา ( คาคินาลิตีของนักเรียน คือ (1,7) )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 รูปแบบของฐานข้อมูล

### 2.5.1 ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

เป็นรูปแบบฐานข้อมูลที่เข้าใจง่ายสำหรับผู้ใช้งานซึ่งโครงสร้างนี้เป็น การจัดเก็บข้อมูลของ เอนทิตีในรูปแบบตารางสองมิติ คือ เป็นแถว (Row) และ เป็นหลัก (column) โครงสร้างแบบนี้ จะทำให้ผู้ใช้งานมองเห็นข้อมูลในลักษณะแบบตาราง โดยไม่ต้องสนใจว่าข้อมูลจะถูกเก็บจริงใน ลักษณะใด และสามารถกำหนดความสัมพันธ์ระหว่างตารางได้ ในการเชื่อมโยงระหว่างตาราง จะเชื่อมโยง โดยใช้แอททริบิวต์ที่มีอยู่ในทั้งสองตารางเป็นตัวเชื่อมโยงข้อมูลกันความสัมพันธ์ ระหว่างข้อมูลจะเป็น ได้ทั้งแบบ 1 : 1, 1 : N, M : N ก็ได้

### 2.5.2 ฐานข้อมูลแบบลำดับชั้น (Hierarchical Relational)

หรือเรียกอีกอย่างหนึ่งว่า โครงสร้างแบบต้นไม้ (Tree Structure) เนื่องจากมีลักษณะ คล้ายต้นไม้ที่คว่ำหัวลงแถวที่อยู่บนจะเป็นแถวพ่อแม่ (Parent record) และแถวถัดมาซึ่ง เป็นแถวลูก (Child record) แถวพ่อแม่แต่ละแถวสามารถมีลูกได้หลายแถว ในขณะที่แถวลูกแต่ละแถวจะมีแถวพ่อแม่ได้เพียงหนึ่งแถวเท่านั้น ความสัมพันธ์ของข้อมูลใน โครงสร้างแบบนี้อาจ เป็นแบบ 1 : 1 หรือ 1 : N

### 2.5.3 ฐานข้อมูลแบบเครือข่าย (Network Relational)

มีลักษณะคล้าย โครงสร้างแบบลำดับชั้นต่างกันตรงความสัมพันธ์ระหว่างข้อมูล จะเป็น แบบ M : N ข้อดีของ relational ที่ทำให้เป็นที่นิยมอย่างแพร่หลายในปัจจุบันก็คือ ใช้งานง่าย เนื่องจากการแก้ไขเปลี่ยนแปลงข้อมูลจะเสมือนกระทำอยู่บนตาราง โดยผู้ใช้ไม่ต้องเข้าไปยุ่ง เกี่ยวกับการเปลี่ยนแปลงตำแหน่งที่เก็บข้อมูลจริงในเครื่องคอมพิวเตอร์ ในขณะที่ Hierarchical และ Network นั้นการทำงานจะไม่อยู่ในรูปของตาราง แต่ละแถวจะมีคอล์มน์ที่เก็บตัวชี้ (pointer) ไปยังตำแหน่งที่แท้จริงของแถวอื่นๆ ที่สัมพันธ์กันผู้ใช้จะต้องเขียนโปรแกรม เพื่อปรับแก้ตัวชี้ให้ถูกต้องเมื่อมีการเพิ่มหรือลบแถวข้อมูล

## 2.6 ภาษาและโปรแกรมที่ใช้ในการพัฒนา

Visual Basic .net เป็นภาษาที่พัฒนาต่อจาก Visual Basic 6.0 หรือพูดง่ายๆ ก็คือเป็น Visual Basic Version 7 ซึ่งขยายขีดความสามารถที่ Visual Basic เดิมไม่สามารถทำได้ โดยเฉพาะในเรื่องของการเป็นภาษาเชิงวัตถุอย่างแท้จริง (สนับสนุนโครงสร้างของภาษาที่เป็น OOP 100%) ทำให้โครงสร้างภาษาของ Visual Basic.NET นั้นมีความสมบูรณ์มากขึ้น แต่ก็ยังคงสนับสนุนรูปแบบการเขียนแบบเดิมไว้ ในบางส่วนเพื่อความสะดวกสำหรับผู้ที่ย้ายจาก Visual Basic Version ก่อนหน้านี้มาบน Visual Basic.NET

Visual Basic.NET เป็นภาษาหนึ่งที่อยู่ในชุดเครื่องมือ Microsoft Studio.NET โดยจะใช้ IDE (Integrated Development Environment) ร่วมกับภาษาอื่นอีก 3 ภาษา ที่อยู่ในชุดเครื่องมือนี้ ซึ่งได้แก่ Visual C#, Visual C++ และ Visual J#

Visual Basic เป็นโปรแกรมที่ใช้สร้างโปรแกรมประยุกต์ สำหรับระบบปฏิบัติการ Windows

Visual เป็นส่วนที่หมายถึงเมธอดในการติดต่อแบบ graphical user interface (GUI) ซึ่งการสร้างทำได้โดยการเพิ่มอ็อบเจกต์ ลงบนฟอร์มที่ทำหน้าที่ติดต่อกับผู้ใช้ผ่านจอภาพ

Basic เป็นส่วนที่หมายถึงภาษา BASIC (Beginners ALL Purpose Symbolic Instruction Code) โดย Visual Basic ได้เปลี่ยนแปลงจากภาษา BASIC ดั้งเดิมด้วยการเพิ่มประโยคคำสั่งฟังก์ชัน และคีย์เวิร์ด ที่เกี่ยวข้อง โดยตรงกับ GUI

แนวคิดของ Visual Basic โปรแกรมประยุกต์ Visual Basic เป็นการพัฒนาในสภาพแวดล้อมของ windows ซึ่ง แนวคิดพื้นฐานในการทำงานของระบบ Windows ที่สำคัญมี 3 ประการ คือ window, events และ ข่าวสาร (message) โปรแกรมประยุกต์ Visual Basic มีการทำงานแบบ Event-Driven ที่เป็นการประมวลผลตามคำสั่งในแต่ละส่วนเพื่อตอบสนองต่อ event ซึ่ง event เหล่านี้สามารถเปลี่ยนโดยการทำงานของผู้ใช้ข่าวสารของระบบหรือโปรแกรมประยุกต์อื่นหรือภายในโปรแกรมเดียวกัน ลำดับการทำงานของ event จะจัดลำดับ โดยจากการประมวลคำสั่ง

โปรแกรมประยุกต์ Visual Basic การสร้างโปรแกรมประยุกต์ Visual Basic ประกอบด้วยขั้นตอนหลัก 3 ขั้นตอน คือ

การสร้างอินเตอร์เฟซ โดยมีฟอร์มเป็นอ็อบเจกต์พื้นฐานและเป็นที่วางตัว control สำหรับการติดต่อกับผู้ใช้

ตั้งค่าคุณสมบัติ เป็นการกำหนดพฤติกรรมและการทำงานให้กับอ็อบเจกต์ต่างๆ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเขียนคำสั่ง เป็นการควบคุมการประมวลผลผ่าน procedure ที่กำหนด

โครงสร้างของ Project ที่สร้างขึ้น สามารถประกอบด้วยไฟล์และอ็อบเจกต์ ดังนี้ Form Modules (.frm) สามารถเก็บข้อความของฟอร์ม ตัว control ที่มีการกำหนดค่าคุณสมบัติ และเก็บระดับการประกาศค่าระดับฟอร์มของค่าคงที่, ตัวแปร และ procedure ภายนอก, event procedure และ procedure ทั่วไป

Class Modules (.cls) คล้ายกับ form module แต่แตกต่างที่จะไม่มีการรบกวนจากผู้ใช้ทั่วไป ใน class module สามารถสร้างอ็อบเจกต์ของผู้พัฒนาโปรแกรมรวมถึงคำสั่ง เมธอด และ คุณสมบัติ

Standard Modules (.bas) สามารถเก็บการประกาศค่า public และระดับโมดูล ของค่าคงที่ประเภทตัวแปร Procedure ภายนอก และ Public Procedure

Resource File (.res) เก็บไฟล์บิตแม็บ, ข้อความ และข้อมูลอื่นที่สามารถเปลี่ยนแปลงได้ โดยไม่ต้องแก้ไขคำสั่ง เช่น ผู้พัฒนาโปรแกรมวางแผนการสร้างโปรแกรมประยุกต์เป็นภาษาอื่น จะสามารถเก็บข้อความที่ติดต่อกับผู้ใช้ และไฟล์บิตแม็บใน resource file แต่ project สามารถมี resource file เพียง 1 ไฟล์ต่อ project

ActiveX Documents (.dob) คล้ายกับฟอร์ม แสดงด้วย internet browser โดย Visual Basic ชุด Professional และ Enterprise สามารถสร้าง ActiveX document

User Control และ Property Page Modules โดย user control (.ctl) และ property page (.pag) เป็น โมดูลที่คล้ายกับฟอร์ม แต่ใช้ในการสร้างตัว control ประเภท ActiveX และ property page ที่ใช้งานร่วมกัน สำหรับการแสดงคุณสมบัติของการออกแบบ Visual Basic ชุด Professional และ Enterprise สามารถสร้าง ActiveX control ได้

Components หมายถึงไฟล์ หรือโมดูลที่เพิ่มเข้าไปใน project ซึ่ง components หลายประเภทที่สามารถเพิ่มเข้าไปใน project

ActiveX Control (.ocx) เป็นตัว control เพิ่มเติมที่สามารถเพิ่มเข้าไปใน toolbox และ ฟอร์ม เมื่อมีการติดตั้ง Visual Basic ไฟล์ที่เก็บตัว control ของ Visual Basic จะได้รับการคัดเลือกไปยังไคลเรคทอรีร่วม (Windows/System) นอกจากนี้ตัว control ประเภท ActiveX มีเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แหล่งที่กว้างขวาง และผู้พัฒนาโปรแกรมสามารถสร้างตัว control ของตัวเองได้ในชุด Professional และ Enterprise

Insertable Objects หมายถึง component เช่น ไฟล์ Excel ที่สามารถสร้างขึ้นเพื่อเป็นส่วนหนึ่งของโปรแกรมในรูปแบบ Integrated Solutions โดยรูปแบบนี้สามารถเก็บข้อมูลที่มีฟอร์แมตที่แตกต่างกัน เช่น ไฟล์กระดาษทำการ (เช่น Excel) บิตแม็บ และข้อความ ซึ่งมาจากโปรแกรมประยุกต์ที่แตกต่างกัน

References เป็นการเพิ่มตัว control ประเภท ActiveX ภายนอก มาใช้ในโปรแกรมประยุกต์ การอ้างถึงทำได้โดยโคออดิเนตของ Reference ซึ่งเข้าถึงด้วย เมนู Reference ในเมนู Project

ActiveX Designer เป็นเครื่องมือในการออกแบบ class ที่ต้องการอ็อบเจกต์ โดยการออกแบบการติดต่อสำหรับฟอร์มที่เป็น designer เริ่มต้น ส่วน designer เพิ่มเติมมีให้จากแหล่งอื่น

Standard Control เป็นตัว control มาตรฐานที่ Visual Basic มีไว้ให้เช่น ปุ่มคำสั่ง combo box โดยจะอยู่ใน toolbox

## 2.6.1 ใน VB.NET มีชนิดของโปรเจกต์ดังต่อไปนี้

Windows Application ใช้พัฒนา Windows Application เราจะสร้างแอปพลิเคชันประเภทนี้ด้วยฟอร์มและคอนโทรล

Class Library ใช้พัฒนา Library ไว้ใช้งาน เป็นชนิดของโปรเจกต์ที่ให้เราสร้างคลาสที่จะใช้ในแอปพลิเคชันอื่นๆ ได้

Windows Control Library เป็นชนิดของโปรเจกต์ ซึ่งจะให้เราสร้างคอนโทรลใหม่ขึ้นมาได้นอกจากคอนโทรลพื้นฐานที่ VB.NET ได้เตรียมไว้ให้

Mobile Web Application ใช้พัฒนา Application บน Mobile Device (ต้องติดตั้ง Mobile Toolkit เพิ่มเติมจึงจะมี Project นี้)

ASP.NET Web Application I ใช้พัฒนา Web Application หรือ Web Form โดยใช้ ASP.NET

ASP.NET Web Service ใช้พัฒนา Web Service ที่สามารถถูกเรียกใช้งานได้จากแอปพลิเคชันอื่นบนอินเทอร์เน็ตเช่นจาก ASP.NET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Web Control Library ใช้พัฒนา Web Control ไว้ใช้งาน ซึ่งจะช่วยให้เราสร้างคอนโทรลใหม่ นอกเหนือจากคอนโทรลที่ VB.NET เตรียมไว้ให้ เพื่อใช้กับแอปพลิเคชัน ASP.NET ได้

Console Application ใช้พัฒนา Console Application (Command Line) แอปพลิเคชันประเภทนี้จะต้องรันในแบบ command - line ซึ่งจะรับอินพุตจากคำสั่งที่เราพิมพ์ลงไป และแสดงผลลัพธ์ออกมาในหน้าต่าง Command Prompt

WindowsService ใช้พัฒนา Windows Service

EmptyProject สร้าง Project ว่างๆ

EmptyWebProject สร้าง Web Project ว่างๆ

New Project in Existing Folder เป็น Project ใหม่ จาก Folder ที่มีอยู่แล้ว

## 2.7 OOP (Object Oriented Programming)

ที่ผ่านมาเรารู้จักการเขียนโปรแกรมแบบ โครงสร้าง ซึ่งเรียนแบบแนวคิดของคนทั่วไป แต่ในระบอบงานที่ซับซ้อนขึ้น การเขียนโปรแกรมแบบโครงสร้าง อาจทำให้เกิดความยุ่งยากมาก ทำให้เกิดมีแนวคิด การเขียนโปรแกรมแบบใหม่ที่เรียกว่า Object Oriented Programming หรือ OO ขึ้นมา ซึ่งช่วยแก้ไขปัญหา การเขียนโปรแกรมที่ซับซ้อนมากขึ้น ถึงแม้ว่า การเขียนโปรแกรมแบบ OOP จะดูเผินๆ เหมือนยุ่งยากขึ้น แต่ก็เป็นการเรียนรู้แบบแนวความคิดของ สิ่งที่พบเห็น ในสิ่งต่างๆ ที่อยู่รอบตัวทั้งนั้น

### 2.7.1 Class และ Object

แนวคิดของ OOP นั้นจะกำหนดให้สิ่งต่างๆ เป็นวัตถุ หรือเรียกว่า Object ซึ่ง Object แต่ละตัว จะถูกสร้างขึ้นมาจาก Class ดังนั้น Object ที่สร้างจาก Class จะถือว่าเป็นคนละตัวกันแต่มีชนิดเดียวกัน เพราะสร้างจาก class เดียวกัน ตัวอย่างเช่น Class ของโทรศัพท์มือถือ ที่สร้างออกมาจากโทรศัพท์มือถือคนละรุ่นกัน เมื่อมองถึง Object ตัวหนึ่งๆ จะเห็นว่ามันมีข้อมูล ที่ใช้กำหนดลักษณะเฉพาะ ของ Object ซึ่งเรียกว่า Property ตัวอย่างเช่น Object ของโทรศัพท์มือถือ จะมี Property ที่แสดง ลักษณะเฉพาะเช่น ยี่ห้อ, สี, รุ่น, จอภาพ และ ภาษา เป็นต้น เมื่อมี Property แล้ว สิ่งหนึ่งที่ขาดไม่ได้ของ Object คือ ความสามารถในการทำงาน ของ Object ซึ่งเรียกว่า Method ตัวอย่างเช่น Object ของโทรศัพท์มือถือ จะมี Switch ปิดเปิด, ปุ่มกดตัวเลข, ระบบการรับส่ง และ เสียงโทรต่างๆ เป็นต้น

เมื่อเราสร้าง Object หนึ่งๆ ขึ้นมาจาก Class ในภาพการเขียนโปรแกรม จะกล่าวได้ว่า

เป็น Object หรือ Instance ของ Class ตัวอย่างเช่น ในการสร้าง Application กับ Delphi เมื่อเราเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อใดๆ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำ Component มาวางลงบน Form เช่น นำ Edit Box ลงมาวางใน Form จะเห็นว่าใน Code Editor นั้น Delphi ได้เขียน Code เพื่อแสดงว่า ได้สร้าง Object Edit ขึ้นมาจาก Class TEdit

### 2.7.2 Member Class

ใน Class ประกอบด้วยข้อมูลต่างๆ นั่นคือ Property และ Method ซึ่งศัพท์ของ โปรแกรมเมอร์เรียกว่า Member ของ Class จุดคืออย่างหนึ่งของ Class คือ การที่ไม่อนุญาตให้ เข้าถึงข้อมูลภายในของ Class ได้โดยตรง จะต้องกระทำผ่าน Member ของ Class สำหรับการ เข้าถึงข้อมูลภายใน Class แบ่งเป็น 3 ระดับ ได้แก่

Private เป็นระดับการเข้าถึงข้อมูล ที่เข้าถึงได้เฉพาะ ข้อมูลที่อยู่ในตัว Class หรือ Instance ของ Class เท่านั้น ไม่อนุญาตให้ผู้ใช้ หรือ Class อื่นเข้าถึงได้ หรือเรียกว่าเป็นแบบ ภายใน

Public เป็นระดับการเข้าถึงข้อมูล ที่เข้าถึงได้จาก ผู้ใช้ Class ทุกคน หรือ Class อื่นๆ โดยไม่จำกัด หรือเรียกว่าเป็นแบบสาธารณะ

Protected เป็นระดับการเข้าถึงข้อมูล เฉพาะ Class ที่ถูกสืบทอด มาจาก Class นี้เท่านั้น ที่เข้าถึงได้ หรือเรียกว่าแบบป้องกันความปลอดภัย

### 2.7.3 Constructor และ Destructor

นอกเหนือจาก Method ที่เรากำหนดให้ Class แล้ว ในการเขียนโปรแกรมแบบ OOP นั้นยังต้องมี Method ที่จำเป็นอยู่อีก 2 ตัว ซึ่งจะต้องทำหน้าที่ ทุกครั้งเมื่อ Object นั้นเริ่มสร้าง ขึ้นมา หรือกำลังจะเลิกใช้งานไป Constructor เป็น Method ที่ทำงานตอน ที่เราเริ่มสร้าง Object ขึ้นมาจาก Class โดยจะทำหน้าที่ที่กำหนด ค่าเริ่มต้น ที่จำเป็นให้กับ Property ต่างๆ จับของ ทรัพยากรณ์ของระบบ ที่จำเป็นต่อการทำงานของ Object ส่วน Destructor เป็น Method ที่ ทำงานในลักษณะตรงกันข้ามคือ ทำหน้าที่ตอน Object นั้นยกเลิกใช้งาน โดยจะคืนทรัพยากร ให้แก่ระบบ

#### 2.7.4 คุณสมบัตินี้ของ OOP

ที่ผ่านมาเป็นการอธิบายแนวความคิด ของการเขียน โปรแกรมแบบ OOP คร่าวๆ ซึ่งการเขียนโปรแกรมแบบ OOP นั้นมีรายละเอียดมากกว่านี้ ซึ่งจะได้กล่าวในตอนต่อไป สำหรับคุณสมบัตินี้ที่สำคัญ ที่ต้องรู้ เพื่อเราจะเขียน โปรแกรมแบบ OOP มี 3 รายการดังนี้

Encapsulation เป็นคุณสมบัตินี้ที่ว่า เราไม่จำเป็นต้องสนใจรายละเอียด ที่ไม่สมควรจะสนใจ เช่น เราเขียนโปรแกรมที่มีการใช้ Button เราก็ไม่ต้องไปสนใจ ถึงขั้นตอนการสร้าง Button ขึ้นมา หรือแม้กระทั่งขั้นตอนการใส่ชื่อ ให้กับ Button ว่ามีขั้นตอนการใส่อย่างไร พุดง่าย อย่าไปสนใจมันเลย เราเพียงแต่กำหนดคุณสมบัตินี้ให้มันก็เพียงพอแล้ว และใช้งานตามความสามารถที่มีเท่านั้น

Inheritance เป็นคุณสมบัตินี้ที่ว่า Class ต้องสามารถสืบทอด ได้เช่นเดียวกับภาษาโปรแกรม ที่กำหนด เป็น Component ทั้งที่ มองเห็นและมองไม่เห็น ก็ต้องสืบทอดได้ โดย คลาสไรฟวดลาส ก็คือ Class ที่ถูกสืบทอดมา นั้น สามารถเพิ่มเติม Property หรือ Method เดิมได้ตามความเหมาะสม

Polymorphism เป็นคุณสมบัตินี้ที่ว่า สามารถเปลี่ยนแปลงความสามารถ ของ Class ให้เข้ากับสภาพแวดล้อมได้ ตัวอย่างง่ายๆ เช่น เราสร้าง Class ที่ชื่อว่า Shape ซึ่งจะใช้สร้าง Object เป็นรูปทรงต่างๆ เช่น วงกลม, สามเหลี่ยม และ สี่เหลี่ยม เป็นต้น แล้วเราก็มี Method Area เพื่อหาพื้นที่ของรูปทรงต่างๆ แน่นอนว่า Method Area ของการเรียกใช้งานแต่ละครั้ง ต้องคำนึงด้วยว่า เราระบุ Property ของรูปทรงว่าเป็นรูปทรงอะไร ซึ่งจะทำให้เรามีวิธีการคำนวณหา ที่แตกต่างกันนั่นเอง

#### 2.8 ฐานข้อมูลที่ใช้ ระบบฐานข้อมูลเชิงสัมพันธ์ (SQL Server 2000)

ระบบฐานข้อมูลจะมี DBMS (Database Management System) เป็นตัวจัดการกับฐานข้อมูลค้นหา จัดระเบียบข้อมูล โดยจะมีโปรแกรมแอปพลิเคชันฐานข้อมูลในการวิ้วข้อมูลปรับเปลี่ยน แก้ไข บ้อน-ลบข้อมูล โปรแกรมแอปพลิเคชันฐานข้อมูลจะเขียนด้วยภาษาโปรซีเจอร์ (Procedural Languages) เช่น cobal, pascal, c หรือ basic เรียกว่าเป็นภาษาในยุคที่ 3 (3GLs = Third Generation Language )

โมดูลของระบบฐานข้อมูลระบบฐานข้อมูลที่ใช้ทำงานอยู่บนเครื่องคอมพิวเตอร์จะมีอยู่หลายโมดูลด้วยกันขึ้นอยู่กับการทำงาน และความเหมาะสม ตลอดจนความสามารถในกาทำงานของเครื่องคอมพิวเตอร์โดยจะมีดังนี้

### 2.8.1 ส่วนประกอบของฐานข้อมูล

ภาษาฐานข้อมูล SQL จะทำการกำหนดโครงสร้างของข้อมูล จัดการข้อมูล คัดสรรข้อมูล โดยการติดต่อและควบคุมระบบจัดการฐานข้อมูล (DBMS) อีกที่หนึ่ง ทางทฤษฎีแล้วภาษาฐานข้อมูลจะมีส่วนประกอบสำคัญอยู่ 3 ส่วน

ส่วนภาษานิยามข้อมูล (Data Definition Language : DDL) เป็นส่วนของคำสั่งที่ใช้กำหนดโครงสร้างข้อมูล สร้างฐานข้อมูล การสร้างตาราง (Base Table) การกำหนดดัชนี (Index) การสร้างวิว (View) เพื่อผลลัพธ์ นอกจากนี้ยังใช้ในการกำหนดกฎเกณฑ์ต่างๆ ให้ DBMS ใช้ในการตรวจสอบความถูกต้องของข้อมูลด้วย

ส่วนของภาษาจัดการข้อมูล (Data Manipulation Language: DML) เป็นส่วนของคำสั่งการค้นหาข้อมูล (Retrive) เพิ่มเติมข้อมูล (Insert) เปลี่ยนแปลงแก้ไขข้อมูล (Update) การลบข้อมูล (Delete)

ส่วนภาษาในการควบคุมข้อมูล (Data Control Language : DCL) เป็นส่วนของคำสั่งที่ใช้ควบคุมการเข้าถึงข้อมูล (Access) ของผู้ใช้ โดยจะกำหนดว่าผู้ใช้คนใดสามารถเพิ่มลบ แก้ไข เปลี่ยนแปลงข้อมูล ผู้ใช้คนใด ทำได้เพียงเรียกดู ค้นหาข้อมูลเป็นการกำหนดสิทธิ์ (Privileges) ให้แก่ผู้ใช้งาน เพื่อความปลอดภัยให้กับข้อมูล

### 2.8.2 Microsoft SQL Sever

ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS) ตัวเก่งของไมโครซอฟต์ที่ได้พัฒนามาจาก Sybase ซึ่งมีความสามารถและประสิทธิภาพในการทำงานสูงสำหรับเครื่องเซิร์ฟเวอร์หลายแพลตฟอร์มด้วยกัน (Intel , Alpha, Power PC, MIPS) Microsoft SQL Sever เป็นผลิตภัณฑ์ทางด้านระบบจัดการฐานข้อมูลเชิงสัมพันธ์ที่สนับสนุนการทำ “ Two phase Commit” เพื่อรักษาความเสถียรภาพของข้อมูลระหว่างเซิร์ฟเวอร์หลายๆตัวในระบบ โดยจะมีกลไกในกา Commit เพื่อเก็บการเปลี่ยนแปลงข้อมูลที่รับส่ง ระหว่างเซิร์ฟเวอร์ จะต้องถูกบันทึกลงในฐานข้อมูลของทั้งสองเครื่องอย่างถูกต้องพร้อมๆกันมิเช่นนั้นให้สามารถทำการ Rollback (ยกเว้นการเปลี่ยนแปลงทั้งหมด) กลับสู่สภาพเดิมมีเครื่องมือสำหรับเขียนสคริปต์ Transact-SQL สามารถจะทำงานได้ทั้งบนเครื่องเซิร์ฟเวอร์และไคลเอ็นต์นอกจากนี้ยังมี SQL/W ( Interactive SQL) สามารถจะทำงานได้ทั้งบนเครื่องเซิร์ฟเวอร์และไคลเอ็นต์นอกจากนี้ยังมี SQL Enterprise Manager เป็นโปรแกรมที่ช่วยทำงานในหลายๆด้าน เช่น สร้างดีไวซ์ อ็อบเจกต์ คอนฟิกูเรชัน กำหนดสิทธิ์ให้ผู้ใช้ Microsoft SQL Sever สามารถจะทำการเรพลิเคตข้อมูลไปสู่ระบบฐานข้อมูลอื่น เช่น ORACLE ACCESS โดยผ่านไดรฟ์เวฟร์ ODBC (open database connectivity) ทำให้สโลแกน “Information at your fingertips” ของไมโครซอฟต์เป็นจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มี MS DTC (Microsoft Distributed Transaction Coordinator) ที่ช่วยจัดการรูทีนของ transact-sql ในการอับเดทข้อมูลของเซิร์ฟเป็นจริง แล้วยังมี MS Transact-SQL ในการอับเดทข้อมูลของเซิร์ฟเวอร์ หลายๆตัวได้ภายในทรานส์แอคชันเดียวกัน

### 2.8.3 ส่วนประกอบพื้นฐานของฐานข้อมูลเชิงสัมพันธ์

ระบบฐานข้อมูลเชิงสัมพันธ์มีส่วนประกอบที่ใช้ในการทำงานคือ ตาราง ซึ่งภายในจะมีทั้งโลว์และคอลัมน์ รวมเป็นเรคอร์ด แล้วยังมีคีย์หลัก-คีย์นอก ซึ่งสร้างจากคอลัมน์ ตลอดจนโดเมนที่ใช้กำหนดขอบเขต-ประเภทของข้อมูล

ตาราง(Table)ฐานข้อมูลเชิงสัมพันธ์จะบรรจุตารางไว้จำนวนมาก และในฐานข้อมูลจะเก็บอินฟอร์มชัน ที่สัมพันธ์กันทุกตาราง ในการเก็บข้อมูลทั้งในคอลัมน์และโลว์จะมีหลักการดังนี้ตารางจะต้องมีชื่อ ไม่ซ้ำไม่ซ้ำกัน (Unique name) และประกอบไปด้วยโลว์และคอลัมน์ทุกคอลัมน์จะบรรจุข้อมูลได้เพียงชนิดเดียว เช่น char, Integer ข้อมูลในแต่ละโลว์จะต้องไม่ซ้ำกัน คอลัมน์(Column) เป็นที่เก็บรายละเอียดต่างๆ ของข้อมูล และกำหนดประเภทของข้อมูลที่จะใช้งานลงไป เช่น คอลัมน์ NAME ให้เป็นประเภทของข้อมูลเป็น Integer/Numeric ในตารางจะประกอบไปด้วยหลายคอลัมน์เพื่อบรรจุรายละเอียดของชื่อที่อยู่ ตำแหน่ง เงินเดือน ประวัติการทำงาน

โลว์ (Row)จะ ประกอบไปด้วยหลายคอลัมน์ คือใน 1 โลว์จะมีข้อมูลต่างๆ ที่อยู่ในแต่ละคอลัมน์ เช่น รหัสพนักงาน ชื่อ-นามสกุล ตำแหน่ง เงินเดือน วันลาป่วย ประวัติพนักงาน และใตารางข้อมูลจะ ประกอบไปด้วยหลายโลว์เช่นกัน

คีย์หลัก (Primary Key)

คีย์หลักหมายถึงคอลัมน์ที่มีข้อมูลไม่ซ้ำกัน (Unique) เป็นเรื่องสำคัญในระบบฐานข้อมูลคอลัมน์ที่มีคุณสมบัติเป็นคีย์หลักได้จะต้องมีค่าของข้อมูลไม่ซ้ำกันเลข เช่นคอลัมน์รหัสพนักงานซึ่งตามปกติแล้วพนักงานแต่ละคนจะมีรหัสประจำตัวต่างกันแน่นอน ข้อกำหนดของคีย์หลักมีดังนี้ในทุกตารางจะต้องมีคีย์หลัก(logical)ใน 1 ตารางจะมีคีย์หลักได้เพียง 1 คีย์เท่านั้น ในคีย์หลักค่าของข้อมูลในคอลัมน์จะต้องไม่เป็นค่าว่าง หรือ Null โดยจะต้องกำหนดเป็น Not Null เสมอ

คีย์นอก(Foreign key)จะเป็นคีย์หลัก หรือคอมบิเนชัน (Combination) ของตารางหนึ่ง สัมพันธ์กับคีย์หลักในตารางอื่น และสามารถจะเชื่อมโยงข้อมูลระหว่างกันได้ดังรูปที่ 1.4 จะเห็นว่าตาราง Personal มีคีย์หลักชื่อ Department เป็นคีย์นอกซึ่งจะใช้ในเชื่อมโยงข้อมูลกับคีย์หลัก Department ในตาราง Location คีย์นอกสามารถจะมีค่าว่าง (Null) ได้

ตาราง Personal

ID.	Name	Address	Position	Department
001	สิงหา	-	ผจก.	100A
002	พรทิพย์	-	เลขา	100A
003	สมชาย	-	วิศวกร	100C
004	ประเสริฐ	-	วิศวกร	100F

ตาราง Location

Department	จังหวัด	ผจก. ส่วนภูมิภาค	รถประจำตำแหน่ง
100 0	กรุงเทพ	บัณฑิต	Merzedezt Benze
100 A	กรุงเทพ	สิงหา	Jeep Charokee
100 B	พิษณุโลก	ไพรวลัย	BMW Si 5
100 C	เชียงใหม่	พชัย	Land Rower
100 D	โคราช	เอกชัย	Pajero
100 E	ประจวบ	เสกสรรค์	Jeep Country

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบระบบ

#### 3.1 วิเคราะห์ระบบ

ระบบการจัดการแพปลา(Fishery Management System) โปรแกรมที่จัดทำขึ้นเพื่อเอื้อประโยชน์กับผู้ประกอบการ โดยการที่จะเข้าไปจัดการกับข้อมูลต้องทำการ Login เข้าไปในระบบ โดยทำงานของโปรแกรมนั้นมีการเก็บรวบรวมข้อมูลต่างๆ อาทิ เช่น ข้อมูลของสมาชิก ข้อมูลเรือ ข้อมูลลูกเรือ ข้อมูลปลา มีการตรวจสอบและแก้ไขข้อมูลที่ถูกต้องตลอดเวลา

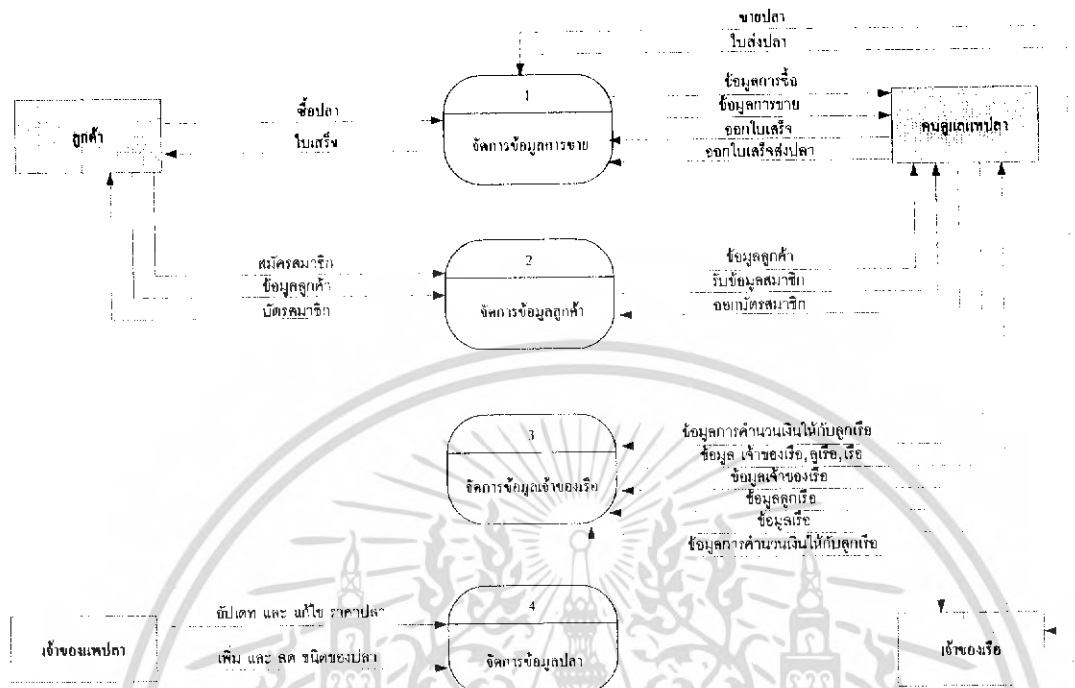
ข้อมูลของเรือ จะเก็บชื่อเรือ ที่อยู่เรือ เจ้าของเรือ วันเวลาที่ออกจท่า และเวลาที่เข้าท่า จำนวนของปลาที่สามารถจับได้ในกรออกหาปลาแต่ละครั้ง รวมถึงการคิดค่าใช้จ่ายในเรือโดยมีค่าของค่าอาหาร ค่าเชื้อเพลิง ค่าซ่อมบำรุงรักษา

ข้อมูลของลูกเรือ จะเก็บชื่อ ที่อยู่ เบอร์โทรศัพท์ ตำแหน่ง เรือที่สังกัด

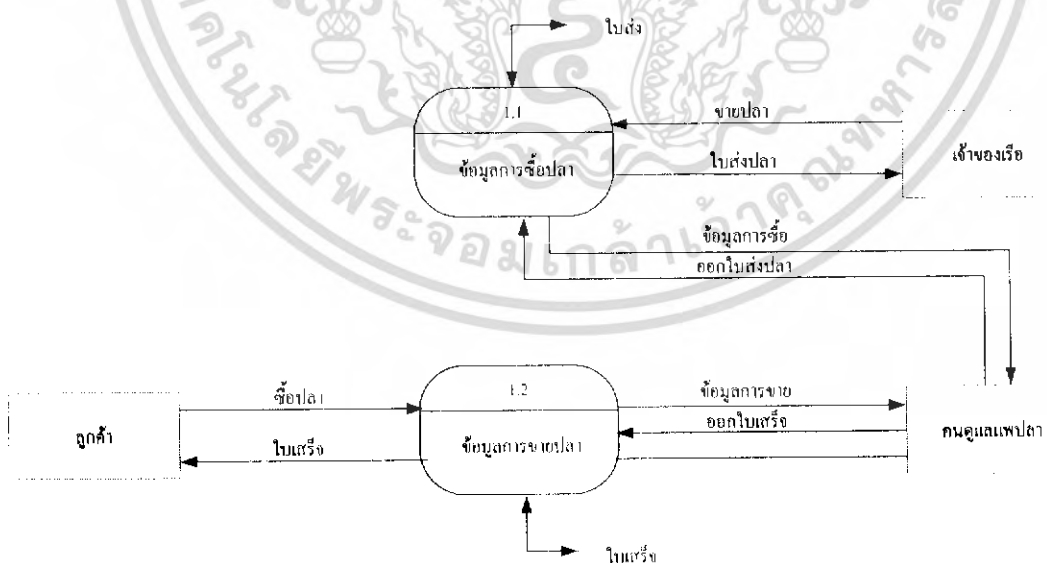
ข้อมูลของปลา จะเก็บข้อมูลของราคาที่ซื้อจากเรือหาปลาโดยตรง ราคาขายให้ลูกค้า ราคาของปลาแต่ละชนิดต่อกิโลกรัม

ข้อมูลของสมาชิก จะเก็บ ชื่อสมาชิก ที่อยู่ เบอร์โทรศัพท์ ชนิดของสมาชิก แบบธรรมดา และวีไอพี



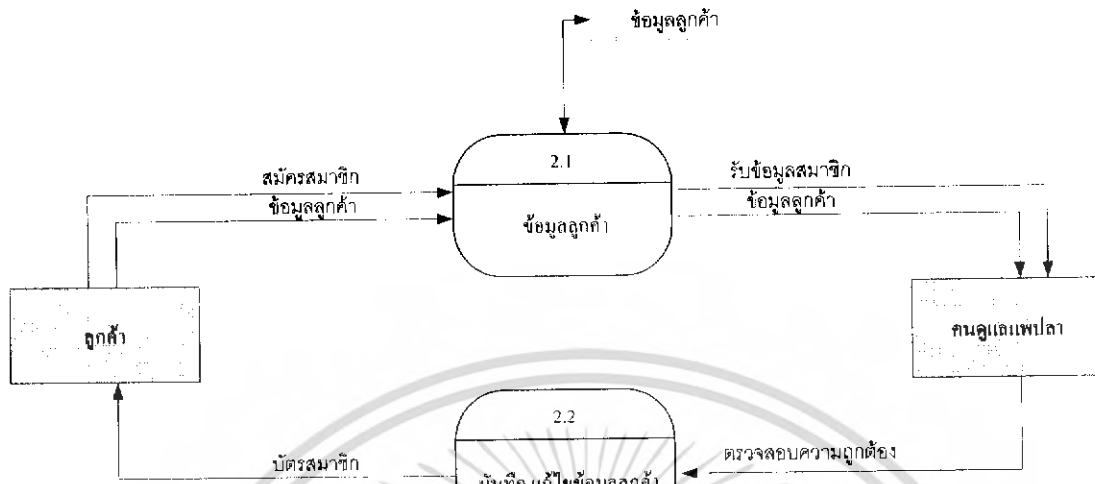


รูปที่ 3.2 Data Flow Diagram Level 1



รูปที่ 3.3 Data Flow Diagram Level 1 ส่วนที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

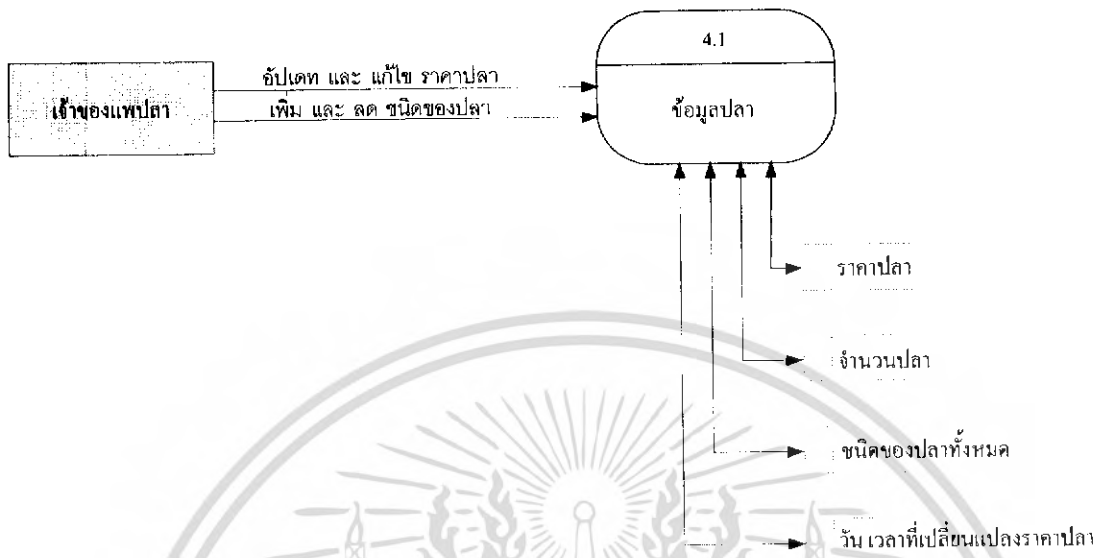


รูปที่ 3.4 Data Flow Diagram Level 1 ส่วนที่ 2



รูปที่ 3.5 Data Flow Diagram Level 1 ส่วนที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 Data Flow Diagram Level 1 ส่วนที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



### 3.4 ดาต้าดิคชันนารี (Data Dictionary)

ตาราง 3.1 สำหรับเก็บสาขา

Name	Type	Key	Null	Meaning
BranchID	Number	PK	no	รหัสสาขา
BranchName	Text	-	No	ชื่อสาขา

ตาราง 3.2 สำหรับเก็บประเภทสินค้า

Name	Type	Key	Null	Meaning
CategoryID	Number	PK	no	รหัสประเภทสินค้า
CategoryName	Text	-	No	ชื่อประเภทสินค้า

ตาราง 3.3 สำหรับเก็บจำนวนสินค้าทั้งหมด

Name	Type	Key	Null	Meaning
CID	Number	PK	no	รหัสสินค้า
BranchID	Number	PK	No	ชื่อสาขา
CInShop	Number	-	No	จำนวน
CAddTypeID	Number	-	No	ประเภท

ตาราง 3.4 สำหรับเก็บรายละเอียดของสินค้า

Name	Type	Key	Null	Meaning
CID	Number	PK	no	รหัสสินค้า
CName	Text	-	No	ชื่อสินค้า
CategoryID	Number	-	No	รหัสประเภทสินค้า
Cost	Number	-	No	ราคาต้นทุน
SalePrice	Number	-	No	ราคาขาย
IsSale	Number	-	No	สถานะของสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.5 สำหรับเก็บสีของเรือ

Name	Type	Key	Null	Meaning
Color	Text	PK	no	สี

ตาราง 3.6 สำหรับเก็บรายชื่อสมาชิก

Name	Type	Key	Null	Meaning
MemberID	Number	PK	no	รหัสสมาชิก
FullName	Text	-	No	ชื่อสินค้า
Sex	Text	-	No	เพศ
IDCard	Number	-	No	รหัสประจำตัวประชาชน
Address	Text	-	No	ที่อยู่
Telephone	Number	-	No	เบอร์โทรศัพท์
MemberTypeID	Number	-	No	รหัสประเภทสมาชิก
BranchID	Number	-	No	รหัสสาขา
AdmitDate	Date	-	No	วันสมัครสมาชิก
ExpireDate	Date	-	No	วันหมดอายุสมาชิก
IsMember	Number	-	No	สถานะการเป็นลูกค้า
IsExpired	Number	-	No	สถานะการเป็นสมาชิก

ตาราง 3.7 สำหรับเก็บรายละเอียดของสินค้า

Name	Type	Key	Null	Meaning
CID	Number	PK	no	รหัสสินค้า
CName	Text	-	No	ชื่อสินค้า
CategoryID	Number	-	No	รหัสประเภทสินค้า
Cost	Number	-	No	ราคาต้นทุน
SalePrice	Number	-	No	ราคาขาย
IsSale	Number	-	No	สถานะของสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.8 สำหรับเก็บประเภทสมาชิก

Name	Type	Key	Null	Meaning
MemberTypeID	Number	PK	no	รหัสสมาชิก
MemberTypeName	Number	-	No	ชื่อประเภทลูกค้า
AdmitMemberFee	Number	-	No	ค่าสมัครสมาชิก
MemberContinue	Number	-	No	ค่าต่ออายุสมาชิก
MemberDuration	Number	-	No	ระยะเวลาการเป็นสมาชิก
BuyRateDiscount	Number	-	No	ส่วนลดเมื่อซื้อสินค้า

ตาราง 3.9 สำหรับเก็บสินค้าที่ซื้อจากเรือ

Name	Type	Key	Null	Meaning
OrderID	Number	PK	no	รหัสสินค้าที่ซื้อ
BranchID	Number	-	No	รหัสสินค้า
OrderDate	Date	-	No	วันที่ซื้อ
ReceiveDate	Date	-	No	วันที่ได้รับ
PaidDate	Date	-	No	วันที่จ่าย
NetDiscount	Number	-	No	ส่วนลด
Net	Number	-	No	ยอดสุทธิ

ตาราง 3.10 สำหรับเก็บรายละเอียดการซื้อ

Name	Type	Key	Null	Meaning
OrderID	Number	PK	no	รหัสสินค้าที่ซื้อ
CID	Number	PK	No	รหัสสินค้า
NumberToOrder	Number	-	No	จำนวนที่ซื้อ
Cost	Number	-	No	ราคา
Discount	Number	-	No	ส่วนลด
Total	Number	-	No	รวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.11 สำหรับเก็บตำแหน่งลูกเรือ

Name	Type	Key	Null	Meaning
Position	Text	PK	no	ตำแหน่ง

ตาราง 3.12 สำหรับเก็บการขายสินค้า

Name	Type	Key	Null	Meaning
SaleID	Number	PK	no	รหัสการขาย
BranchID	Number	-	No	รหัสสาขา
MemberID	Number	-	No	รหัสสมาชิก
SaleDate	Date	-	No	วันที่ขาย
NetTotal	Number	-	No	ยอดรวม
SaleBy	Text	-	No	ขายโดย

ตาราง 3.13 สำหรับเก็บรายละเอียดการขาย

Name	Type	Key	Null	Meaning
SaleID	Number	PK	no	รหัสการขาย
CID	Number	PK	No	รหัสสินค้า
SalePrice	Number	-	No	ราคาขาย
Discount	Number	-	No	ส่วนลด
Amount	Number	-	No	จำนวน
Total	Number	-	No	รวม
MeReceived	Number	-	No	ยอดที่รับ

ตาราง 3.14 สำหรับเก็บขนาดของเรือ

Name	Type	Key	Null	Meaning
Size	Text	PK	no	ขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 3.15 สำหรับเก็บข้อมูลเจ้าของเรือ

Name	Type	Key	Null	Meaning
SupplierID	Number	PK	no	รหัสเจ้าของเรือ
SupplierName	Text	-	No	ชื่อ
Address	Text	-	No	ที่อยู่
ContactName	Text	-	No	ชื่อผู้ติดต่อ
Telephone	Number	-	No	เบอร์โทรศัพท์
Color	Text	-	No	สี
Size	Text	-	No	ขนาด

ตาราง 3.16 สำหรับเก็บข้อมูลลูกเรือ

Name	Type	Key	Null	Meaning
SupplierIDBoy	Number	PK	no	รหัสเจ้าลูกเรือ
PositionBoy	Text	-	No	ชื่อ
AddressBoy	Text	-	No	ที่อยู่
BoatBoy	Text	-	No	เรือที่ประจำอยู่
ContactNameBoy	Text	-	No	ชื่อผู้ติดต่อ
TelephoneBoy	Number	-	No	เบอร์โทรศัพท์

ตาราง 3.17 สำหรับเก็บLogin

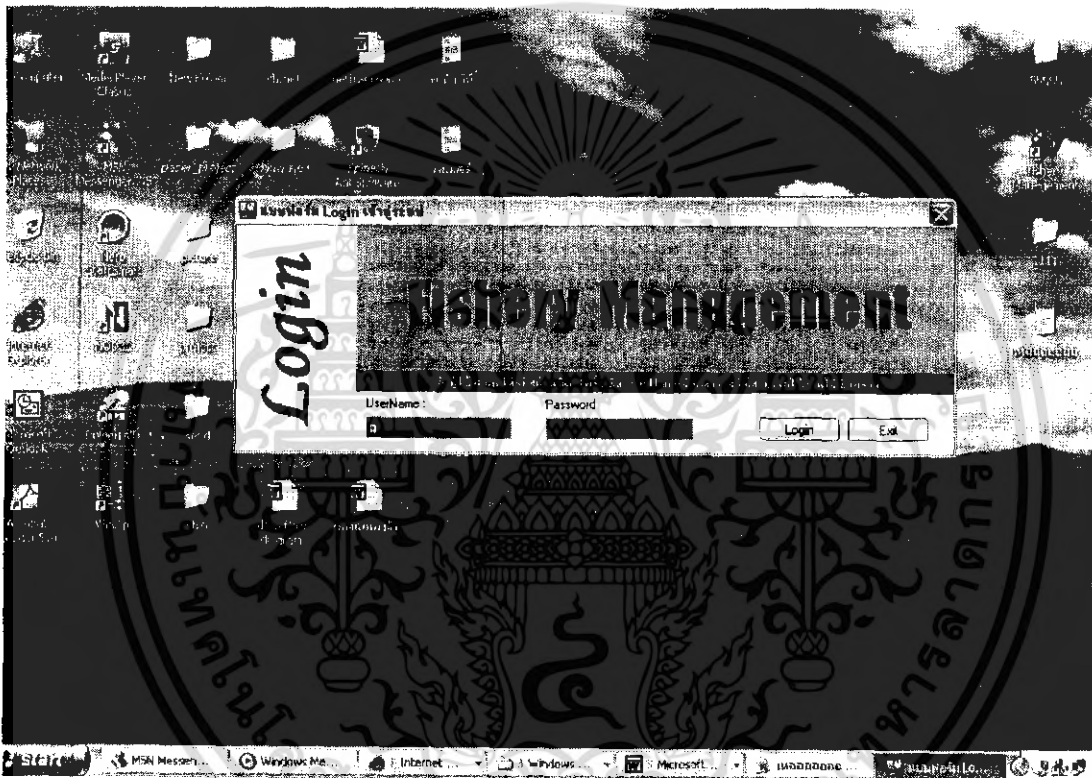
Name	Type	Key	Null	Meaning
UserName	Text	PK	no	UserName
Password	Text	-	No	Password
IsNorinal	Number	-	No	สถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลองและผลการทดลอง

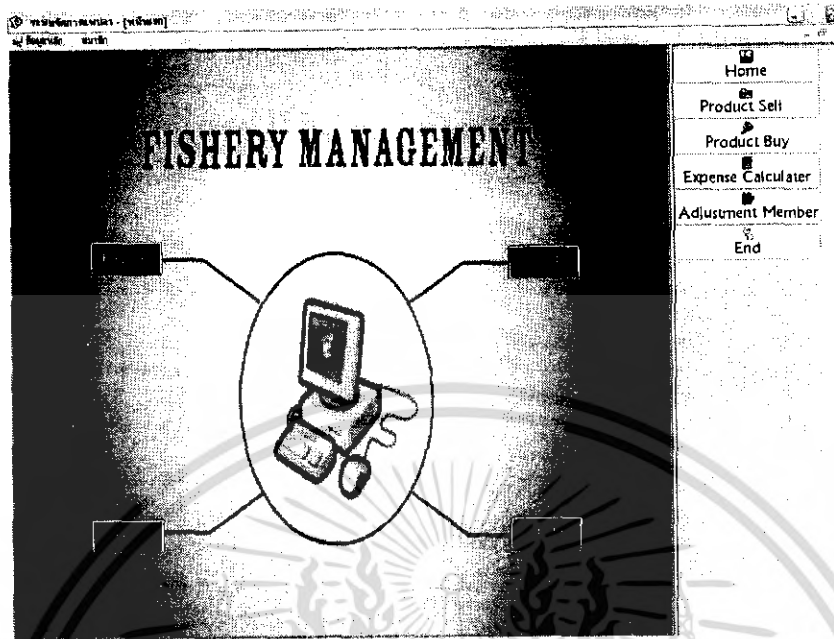
การทำงานของโปรแกรมโดยการติดตั้งไฟล์ setup เมื่อทำการติดตั้งโปรแกรมเสร็จแล้ว การที่โปรแกรมจะสามารถรันหรือทำงานได้ต้องทำการลงโปรแกรมฐานข้อมูล โดยฐานข้อมูลที่ใช้ในโปรแกรมคือ SQL Server 2000 เพื่อเก็บข้อมูลต่างๆ



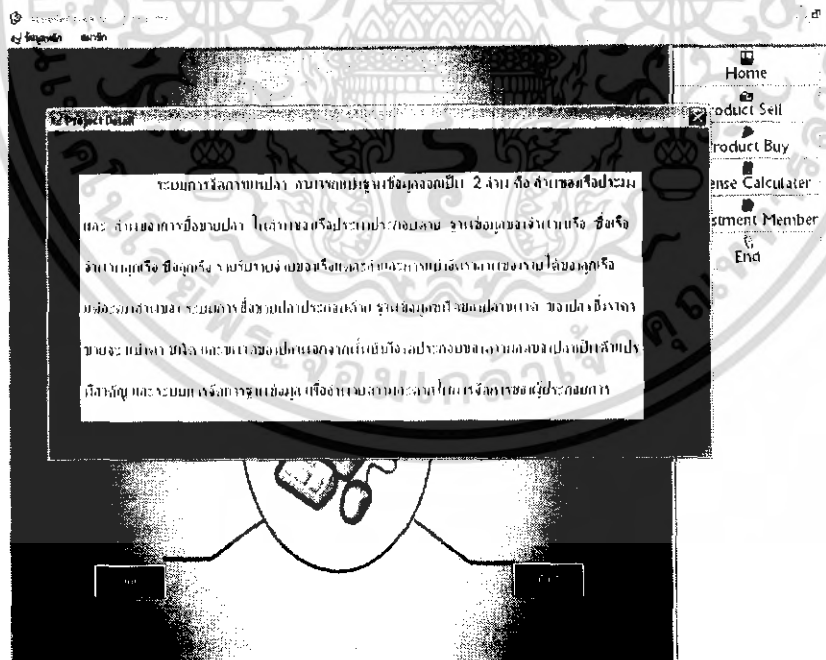
รูปที่ 4.1 แสดงหน้าต่าง Login เข้าระบบ

รูปที่ 4.1 เป็นหน้าที่ทำการ Login เข้าสู่ระบบโปรแกรม โดยต้องมีการใส่ username=q และ password=q

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 หน้าหลักของตัวโปรแกรม  
รูปที่ 4.2 เป็นหน้าโปรแกรมที่จะคลิก ไปยังส่วนต่างๆ

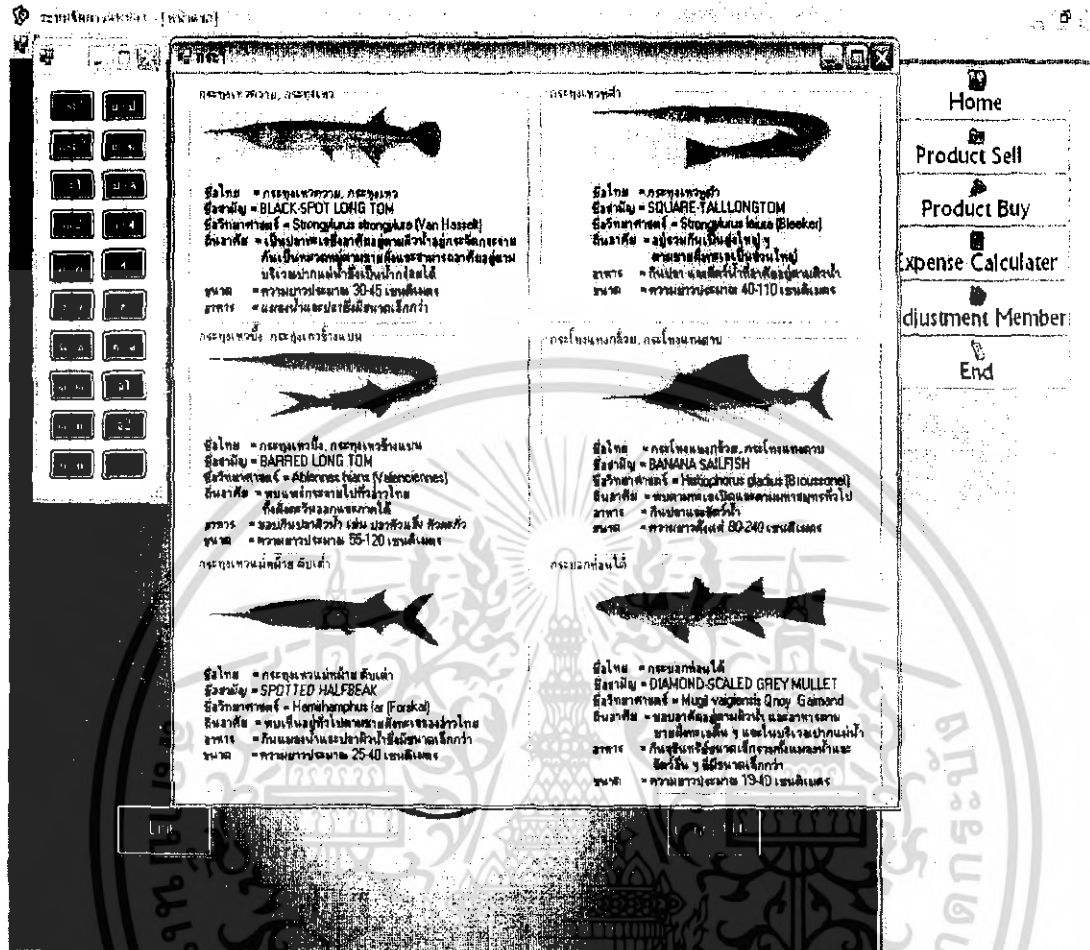


รูป 4.3 แสดงจุดประสงค์ในการทำโปรแกรมระบบจัดการแพปลา

รูปที่ 4.3 จะบอกถึงจุดประสงค์ในการทำโปรแกรมนี้นั้นมาร่วมถึงบอกรายละเอียดต่างๆของ

โปรเจก

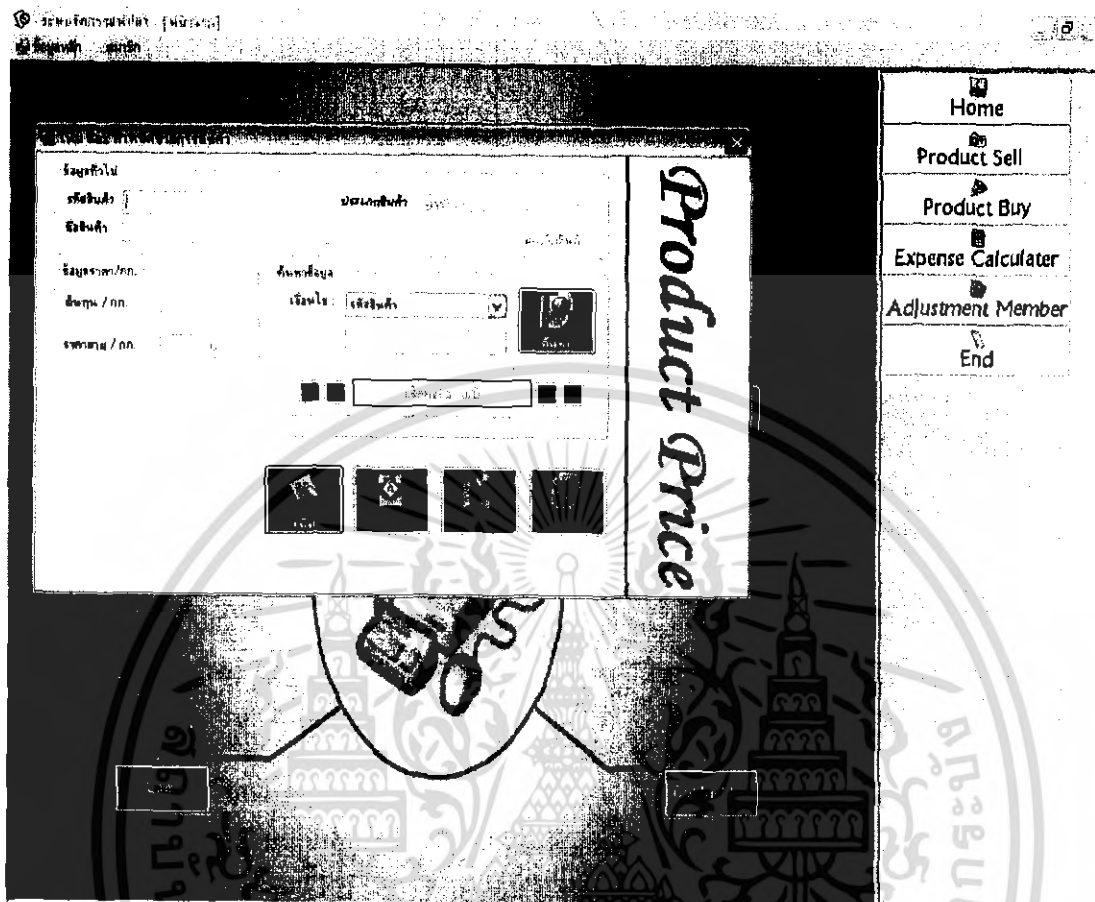
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดงข้อมูลและรายละเอียดของสัตว์น้ำแต่ละชนิด

รูปที่ 4.4 จะบอกรายละเอียดข้อมูลของสัตว์น้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 ส่วนของการกำหนดราคาซื้อจากเรือและราคาที่ขายให้ลูกค้า

รูปที่ 4.5 ในส่วนของตรงนี้จะเป็นส่วนของการกำหนดราคาปลาที่ซื้อมาจากเรือประมงและกำหนดราคาให้กับลูกค้า

การที่จะเปิดหน้าต่างตรงส่วน เพิ่มและกำหนดราคาสินค้านั้นเข้าไปที่เมนูข้อมูลหลักหรือ กด F1

เมื่อต้องการเพิ่มข้อมูลของสินค้าตรงกดตรงปุ่มเพิ่มด้านล่างถึงจะสามารถเพิ่มข้อมูลได้ ข้อมูลทั่วไป

รหัสสินค้าคือ เป็นการป้อนข้อมูลของรหัสสินค้า

ชื่อสินค้า เป็นการป้อนข้อมูลของชื่อปลา

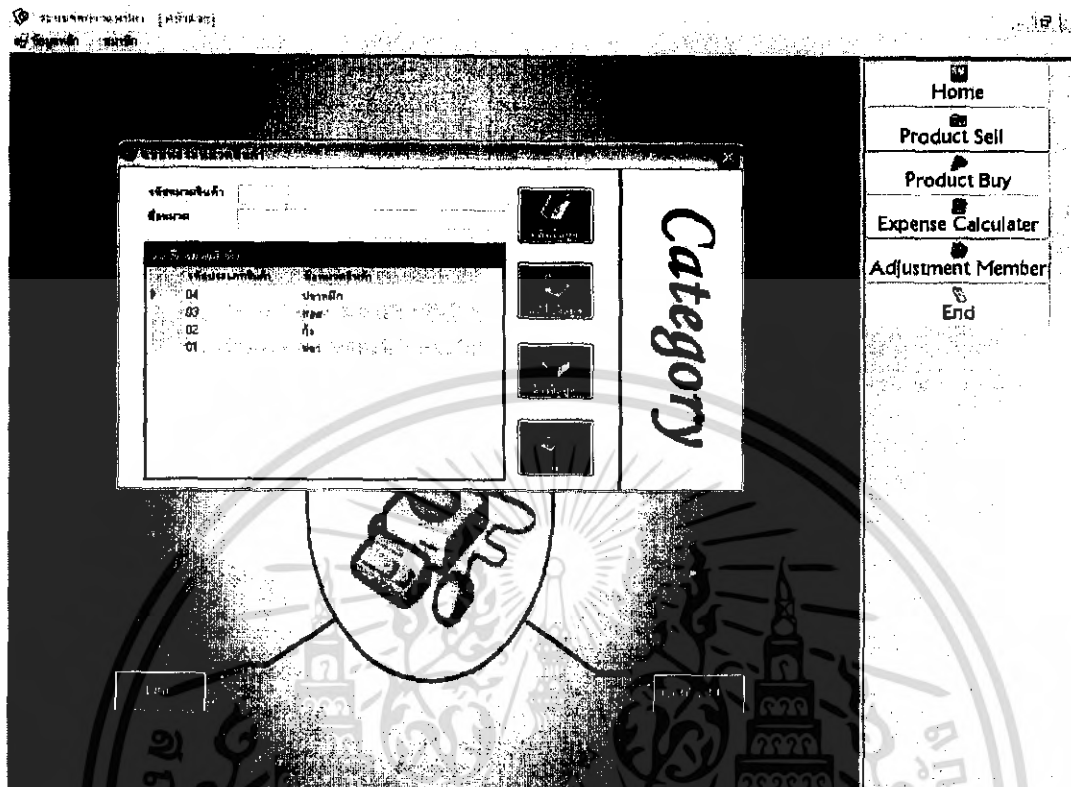
ข้อมูลราคาปลา กก.

ต้นทุน/กก เป็นข้อมูลกำหนดราคาปลาที่รับจากเรือประมง

ราคาขาย/กก เป็นข้อมูลของการกำหนดราคาที่ขายให้กับลูกค้า

ประเภทสินค้า คือเป็นการกำหนดชนิดของข้อมูลอาทิเช่น ปลา กุ้ง หอย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงส่วนของการเพิ่มชนิดข้อมูลของสัตว์น้ำ

รูปที่ 4.6 จะเป็นการกำหนดประเภทของสัตว์น้ำ โดยสามารถเพิ่มข้อมูลของประเภทสัตว์น้ำต่างๆที่ต้องการได้

ในส่วนของหน้าต่างแบบฟอร์มหมวดสินค้าตรงส่วนนี้เข้าไปที่ ข้อมูลหลัก คลิกที่ หมวดสินค้าต่างๆในแพ หรือกดF2

รหัสหมวดสินค้า เป็นการกำหนดประเภทของรหัสสินค้า

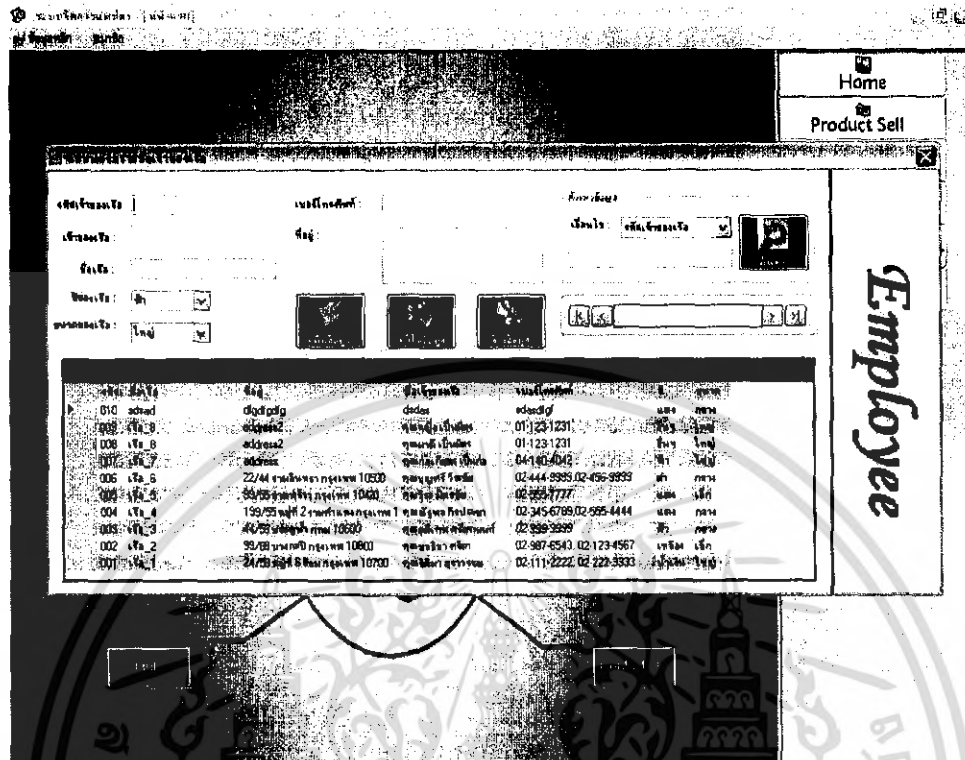
ชื่อหมวด เป็นการกำหนดประเภทของสินค้า

ปุ่ม เพิ่มข้อมูล เป็นการกดเพื่อยืนยันการป้อนข้อมูล

ปุ่ม แก้ไขข้อมูล เป็นส่วนของการแก้ไขข้อมูล

ปุ่ม ถ้างข้อมูล เป็นส่วนของการลบที่ยังไม่บันทึกข้อมูล

ปุ่ม จบ เป็นการจบการทำงานของโปรแกรม



รูปที่ 4.7 แสดงการเก็บข้อมูลรายละเอียดของเรือ

รูปที่ 4.7 จะเป็นการเก็บข้อมูลของเรือ โดยมีการเก็บ รหัสเรือ เจ้าของเรือ ชื่อเรือ สีของเรือ ขนาดของเรือ เบอร์โทร ที่อยู่

ในส่วนของหน้าตาแบบฟอร์มรายชื่อเจ้าของเรือ เมื่อต้องการเข้าไปที่หน้าตานี้ก็เข้าไปที่ข้อมูลหลัก แล้วไปคลิกที่ แบบฟอร์มรายชื่อเจ้าของเรือหรือกด F3

รหัสเจ้าของเรือ เป็นการกำหนดหมายเลขรหัสของเจ้าของเรือ

เจ้าของเรือ เป็นส่วนของการเพิ่มชื่อของเจ้าของเรือ

ชื่อเรือ เป็นส่วนบอกชื่อของเรือ

สีของเรือ กำหนดสีของเรือ

ขนาดของเรือ คือเป็นการเลือกขนาดของเรือ

เบอร์โทรศัพท์ ข้อมูลเบอร์โทรศัพท์

ที่อยู่ เป็นการบอกที่อยู่ของเรือ

ส่วนของการค้นหา มีค้นหาตาม รหัสของเจ้าของเรือ ชื่อเรือ ชื่อ-เรือ

รหัสลูกเรือ	ชื่อ-สกุล	เบอร์โทรศัพท์	ที่อยู่	ตำแหน่ง	สถานะ
001	เ.เ. 2	555	111	111	111
002	เ.เ. 3	555	111	111	111
003	เ.เ. 4	555	111	111	111
004	เ.เ. 5	555	111	111	111
005	เ.เ. 6	555	111	111	111
006	เ.เ. 7	555	111	111	111
007	เ.เ. 8	555	111	111	111
008	เ.เ. 9	555	111	111	111
009	เ.เ. 10	555	111	111	111
010	เ.เ. 11	555	111	111	111
011	เ.เ. 12	555	111	111	111
012	เ.เ. 13	555	111	111	111
013	เ.เ. 14	555	111	111	111
014	เ.เ. 15	555	111	111	111
015	เ.เ. 16	555	111	111	111
016	เ.เ. 17	555	111	111	111
017	เ.เ. 18	555	111	111	111
018	เ.เ. 19	555	111	111	111
019	เ.เ. 20	555	111	111	111

รูปที่ 4.8 แสดงการเก็บข้อมูลรายละเอียดของลูกเรือ

รูปที่ 4.8 จะเก็บข้อมูลของลูกเรือ โดย เก็บรหัสลูกเรือ ชื่อ-สกุล เบอร์โทรศัพท์ ที่อยู่ ตำแหน่ง เรือที่ประจำอยู่

ในส่วนของหน้า قالبฟอร์มรายชื่อลูกเรือ เมื่อต้องการเข้าไปที่หน้า قالبนี้ให้เลือกที่เมนู ข้อมูลหลัก แล้วไปคลิกที่แบบฟอร์มรายชื่อลูกเรือหรือกด F4

รหัสลูกเรือ ป้อนข้อมูลหมายเลขประจำเรือ

เรือที่อยู่ประจำอยู่ กรอกชื่อเรือที่ลูกเรือประจำอยู่

ตำแหน่ง คือกรอกตำแหน่งของพนักงานที่อยู่ในเรือ

ชื่อ-สกุล กรอกข้อมูลชื่อลูกเรือ

เบอร์โทรศัพท์ กรอกข้อมูลเบอร์โทรศัพท์

ที่อยู่ กรอกข้อมูลของลูกเรือ

ค้นหา ในส่วนของการค้นหาโดยค้นหาข้อมูลของ รหัสลูกเรือ ชื่อเรือ ชื่อ-สกุล ตำแหน่ง

รูปที่ 4.9 แสดงส่วนการเก็บข้อมูลของสมาชิก

รูปที่ 4.9 แสดงหน้าของการสมัครสมาชิกโดยมีการกรอกข้อมูลต่างๆที่จำเป็น และเลือกประเภทของสมาชิกโดยแบ่งเป็นสองประเภทคือ สมาชิกแบบธรรมดา และ แบบวีไอพี

เมื่อต้องการใช้งานส่วนของหน้าดังเมนูนี้เข้าไปที่สมัครสมาชิกเลือกคลิกที่สมัครสมาชิกใหม่หรือกด F6

ข้อมูลของลูกค้าที่ต้องการสมัครสมัครเป็นสมาชิกที่ต้องการกรอกข้อมูลมีดังนี้เมื่อต้องการเพิ่มข้อมูลของสมาชิกต้องคลิกที่ปุ่ม เพิ่มรายชื่อ สมาชิกเลขที่ลำดับของสมาชิกจะขึ้นมาโดยจะมีส่วนของการเก็บข้อมูลดังนี้

ชื่อ-สกุล กรอกชื่อและนามสกุลของสมาชิก

เพศ เลือกเพศของสมาชิก

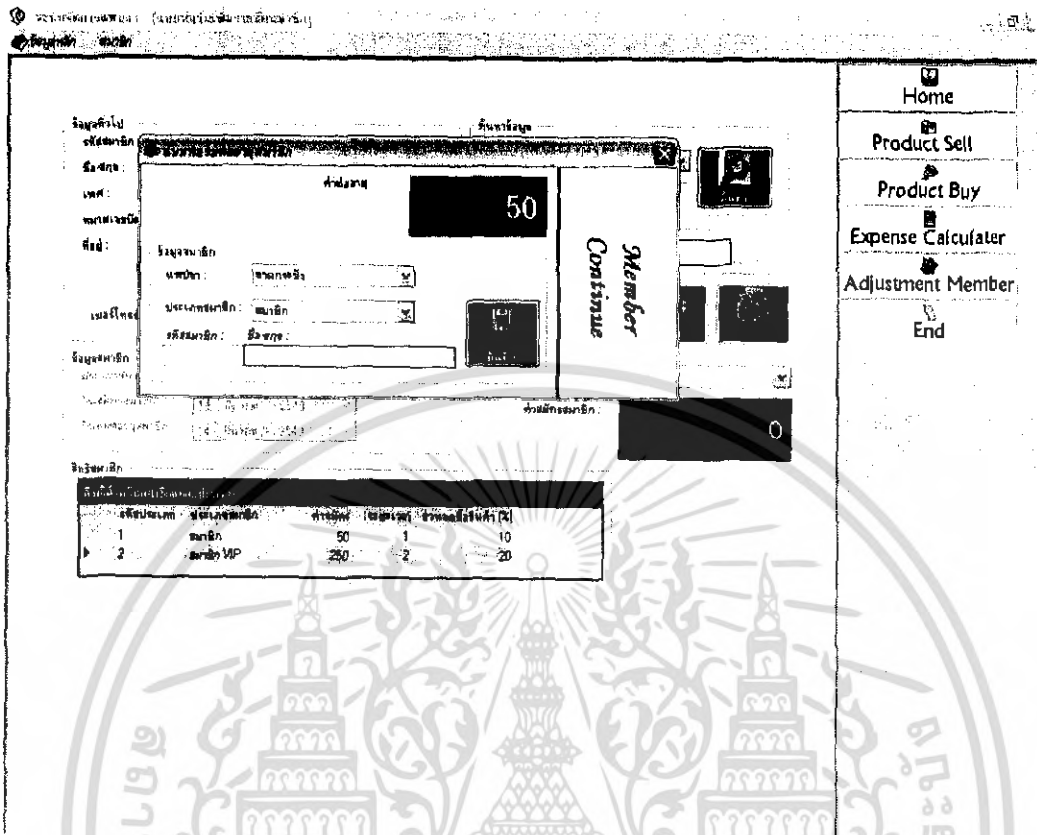
หมายเลขบัตร กรอกเลขที่ประจำบัตรประชาชน

ที่อยู่ ข้อมูลที่อยู่ของสมาชิก

เบอร์โทรศัพท์ กรอกเบอร์โทรศัพท์ของสมาชิก

สิทธิสมาชิก โดยเลือกประเภทของสมาชิกมีสองที่ลูกค้าต้องการสมัคร

ในส่วนของการค้นหาข้อมูลสามารถค้นหาโดย รหัสสมาชิก และชื่อ-สกุล

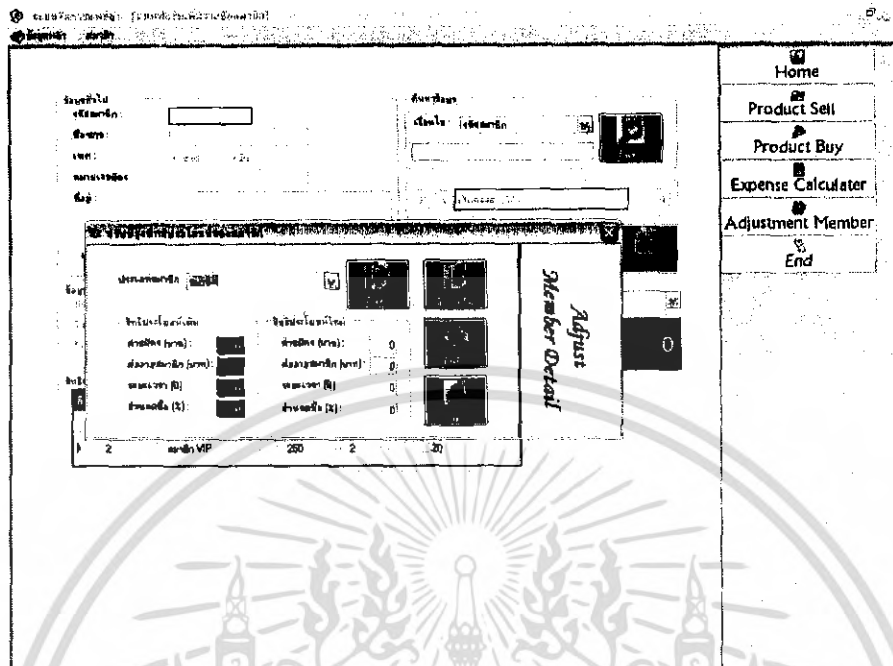


รูปที่ 4.10 แสดงหน้าต่างของการต่ออายุความเป็นสมาชิก

รูปที่ 4.10 เป็นส่วนของการต่อสมาชิกที่หมดอายุ

ในส่วนของหน้าต่างแบบฟอร์มต่ออายุสมาชิก เมื่อต้องการเข้าไปต่ออายุของสมาชิกที่หมดอายุ เราสามารถเข้าไปที่เมนู สมาชิกเลือกคลิกเมาส์ไปที่ ต่ออายุสมาชิกหรือกด F7

เลือกประเภทสมาชิก คือ ในส่วนตรงนี้เป็นกรเลือกสมาชิกมีสองประเภท คือ สมาชิกหรือสมาชิก VIP จะได้รับสิทธิประโยชน์แตกต่างกันรหัสสมาชิก คือการใส่รหัสสมาชิกแล้วเอ็นเตอร์ ชื่อสมาชิกขึ้นจะขึ้นมาปุ่มบันทึก เป็นการยืนยันการต่ออายุ



รูปที่ 4.11 แสดงหน้าต่างของการปรับปรุงแก้ไขสิทธิประโยชน์ของสมาชิก

รูปที่ 4.11 จะเป็นส่วนของการปรับปรุงแก้ไขสิทธิประโยชน์ต่างๆของสมาชิก

ในส่วนของหน้าต่างแบบฟอร์มปรับปรุงสิทธิประโยชน์สมาชิก เมื่อต้องการเข้าไปที่หน้าต่างต้องส่วนนี้ก็เข้าไปที่ เมนู สมาชิก แล้วเลือกคลิก ปรับปรุงสิทธิประโยชน์ของสมาชิกหรือ กด F8

ประเภทสมาชิก เป็นการเลือกสมาชิกที่ต้องการปรับปรุง คือ สมาชิก หรือ สมาชิก VIP โดยเลือกแล้วก็ กดคลิกที่ปุ่ม ค้นหาข้อมูลของระบบเดิก็จะปรากฏขึ้นมาตรงส่วน สิทธิประโยชน์เดิม

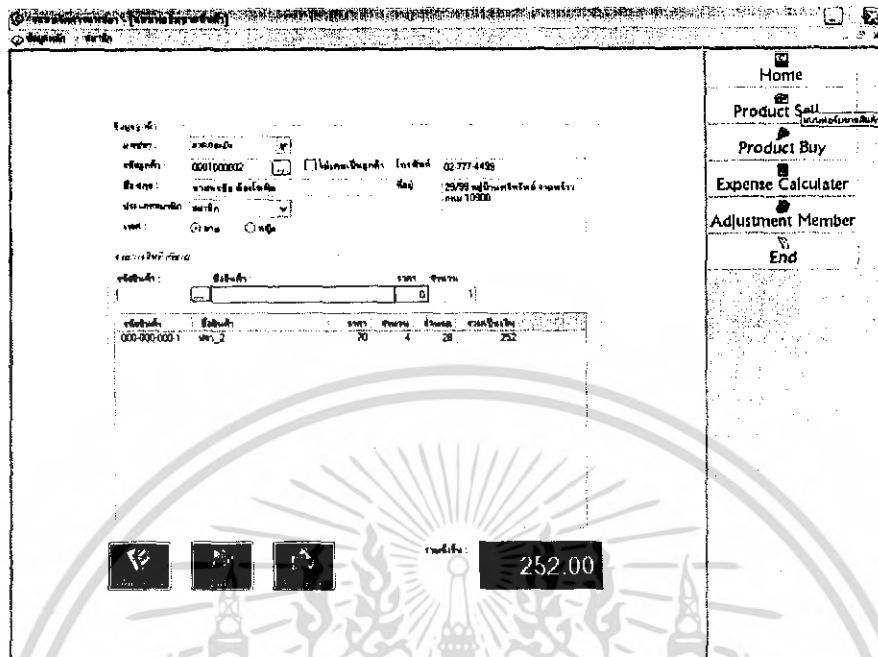
โดยมาเปลี่ยนแปลงข้อมูลตรงส่วนปรับปรุงสิทธิประโยชน์ใหม่ในส่วนของ

ค่าสมัคร(บาท) เป็นส่วนค่าสมัครที่ต้องการเพิ่มหรือแก้ไข

ต่ออายุสมาชิก(บาท) เป็นค่าคิดสำหรับสมาชิกที่ต้องการต่อราคาใหม่

ระยะเวลา(ปี) เป็นระยะเวลาที่กำหนดสิทธิของสมาชิก

ส่วนลดเป็น(%) เป็นส่วนของเปอร์เซ็นต์ที่ลดให้กับสมาชิก



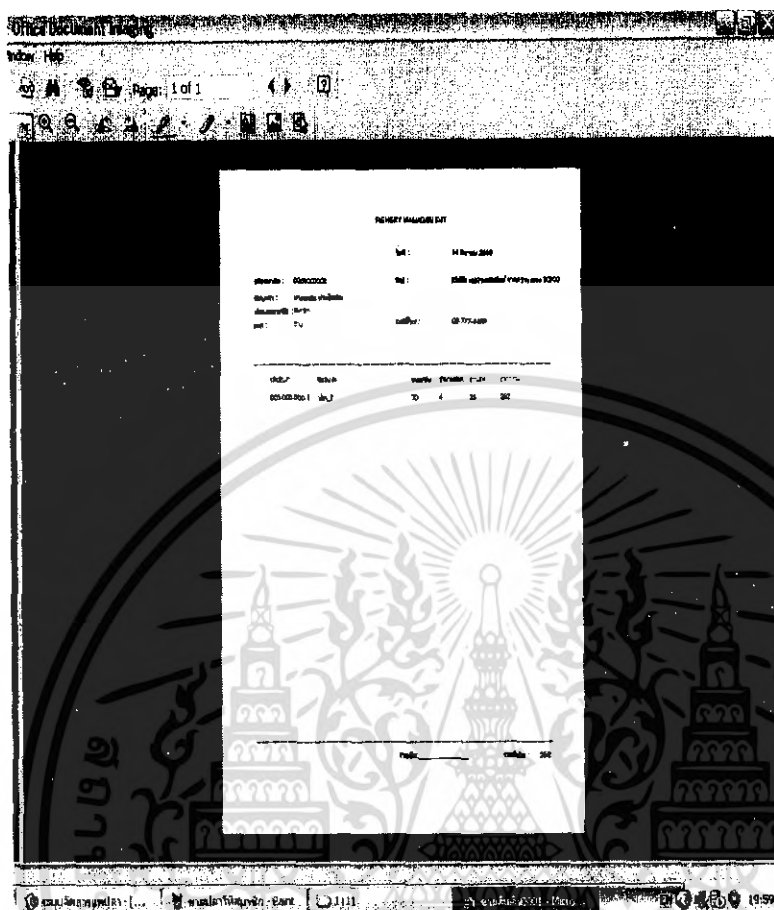
#### รูปที่ 4.12 แสดงหน้าต่างของการขายสัตว์น้ำให้ลูกค้า

รูปที่ 4.12 จะแสดงในส่วนของการขายสัตว์น้ำให้กับลูกค้าโดยคำนวณราคารวม

ในส่วนหน้าต่างการขายสินค้าเป็นส่วนของการขายสินค้าให้กับลูกค้าโดยเข้าไปที่ Product Sell

ในส่วนของการขายปลาให้กับลูกค้าที่ไม่ใช่สมาชิกคือ ต้องไปคลิก ในตรงส่วน ไม่เคยเป็นลูกค้าแล้วก็เพิ่ม ชื่อ ที่อยู่ เบอร์โทรศัพท์ การเพิ่มข้อมูลสินค้า สามารถเลือกของรหัสสินค้าโดยเลือกสินค้าที่ลูกค้าต้องการ ก็จะขึ้นชื่อสินค้า ราคา โดยกำหนดจำนวนของปลาเป็นกิโลกรัมและก็เพิ่มข้อมูลที่เราต้องการ โดยการกดต้อง ปุ่ม เพิ่มรายการเมื่อเพิ่มรายการเสร็จทำการบันทึกข้อมูลก็จะเก็บไว้และก็จะปริ้นใบเสร็จออกมา

ในส่วนของสมาชิกก็สามารถแอดชื่อของสมาชิกขึ้นมาจากฐานข้อมูลและก็จะแอดข้อมูลของปลาและก็เพิ่มรายการข้อมูลที่จะแอดลงฐานข้อมูล



รูปที่ 4.13 แสดงใบเสร็จออกให้กับลูกค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.14 แสดงหน้าต่างของการรับข้อมูลของสัตว์ จากเรือประมง

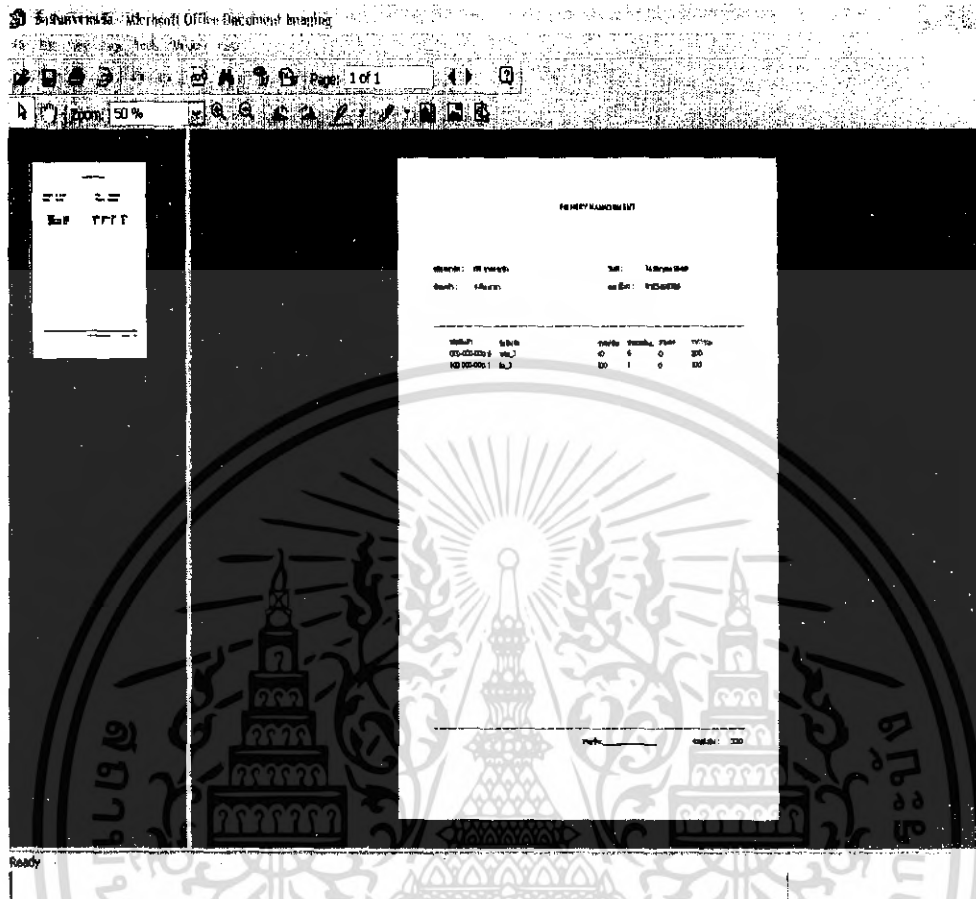
รหัสสินค้า	ชื่อสินค้า	จำนวน	จำนวน	จำนวน	รวมเป็นเงิน
000-0000	ปลา_3	40	5	0.00	200.00
100-0000	ไก่_3	100	1	0.00	100.00

รวม: 0.00    รวมเป็นเงิน: 330.00

รูปที่ 4.14 แสดงหน้าต่างของการรับข้อมูลของสัตว์ จากเรือประมง

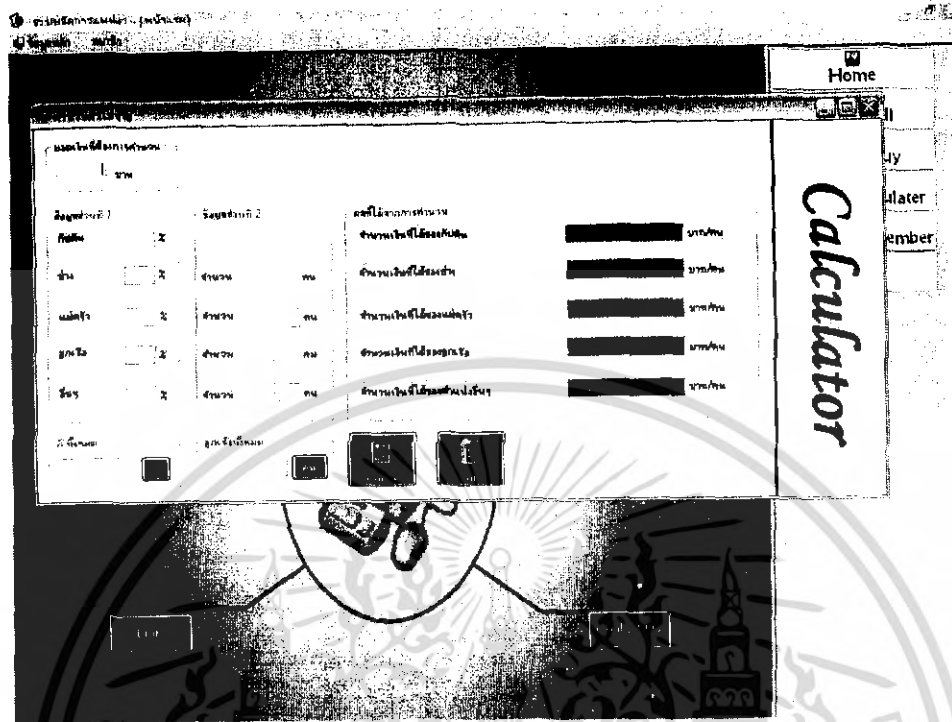
รูปที่ 4.14 ในส่วนของหน้าต่างการรับซื้อสัตว์น้ำจากเรือประมง โดยมีการคิดราคารวม

ในส่วนของหน้าต่างตรงส่วนนี้เป็นส่วนของการที่เรือประมงนำปลามาส่งยังแพปลาโดยหน้าต่างนี้สามารถไปเลือกคลิกได้ที่ Product Buy จะมีส่วนของข้อมูลชื่อเรือ ชื่อคนที่นำสินค้ามาขาย วันที่ เบอร์โทรศัพท์ รายการสินค้าสามารถแอดเพิ่มได้จากฐานข้อมูล เมื่อเพิ่มรายการสินค้าเสร็จ ก็คิดตรง ปุ่ม เพิ่มรายการสินค้าข้อมูลจะไปเก็บที่ฐานข้อมูลและทำการปรีนใบเสร็จออกมา



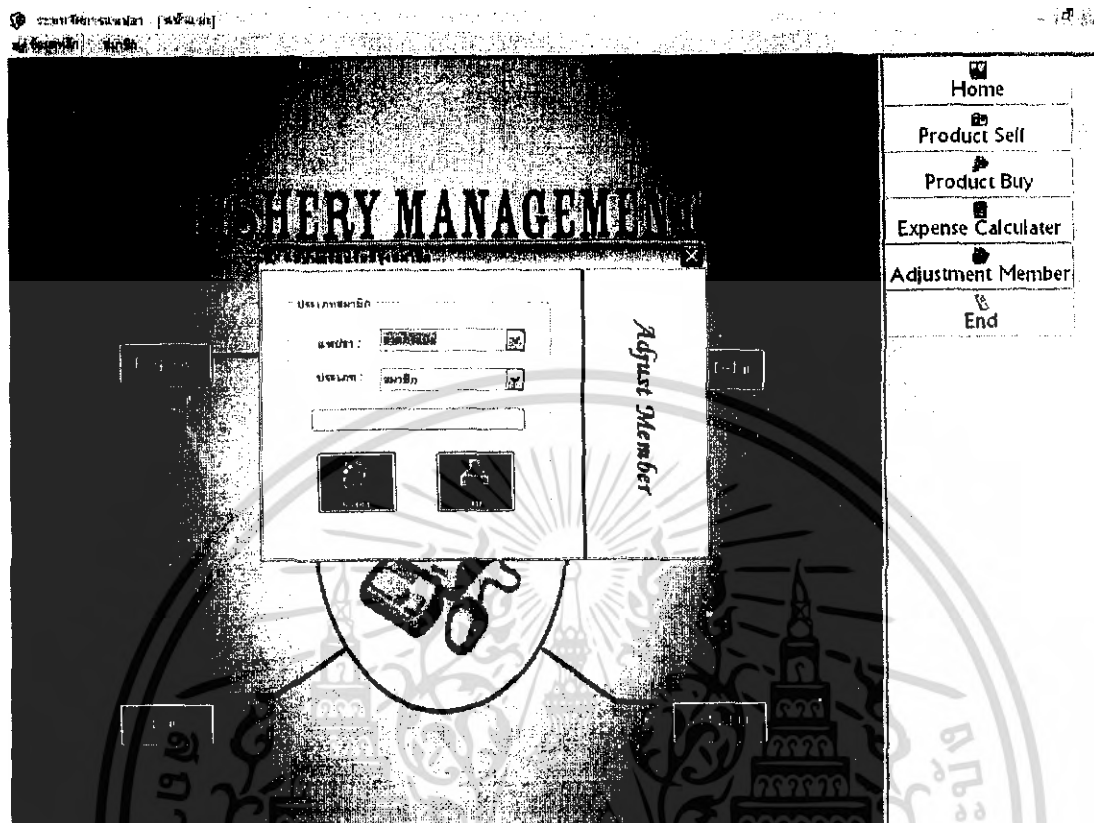
รูปที่ 4.15 แสดงข้อมูลใบเสร็จที่ออกให้กับเรือประมง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 แสดงการคิดคำนวณรายได้ให้กับลูกเรือ

รูปที่ 4.16 เป็นส่วนของการคิดส่วนแบ่งให้กับลูกเรือ โดยคิดเป็นส่วนแบ่งโดยเปอร์เซ็นต์จากรายได้  
 ในส่วนของหน้าตาตรงคำนวณรายได้ของลูกเรือแต่ละคนนั้นเราสามารถเลือกหน้าตาต่างนี้  
 ได้โดยคลิกที่ Expense Calculator โดยจะมีข้อมูลมาให้กรอกคือ ข้อมูลของ ยอดเงินที่ต้องการ  
 คำนวณ โดยใส่ยอดเงินที่ต้องการแบ่งให้กับลูกเรือ โขมีการคิดส่วนแบ่งเป็นเปอร์เซ็นต์ของรายได้รวม  
 ทั้งหมด โดยมีช่องให้กรอกข้อมูลเป็น% โดยข้อมูลที่กรอกต้องรวมกันได้ครบ 100% ส่วนกรณีของ  
 ลูกเรือก็ใส่จำนวนของพนักงานในเรือมีตำแหน่งละกี่คนเมื่อใส่เสร็จ ก็สามารถคำนวณได้โดยการกด  
 ปุ่มด้านล่างจะทำการคำนวณหา รายได้ของลูกเรือแต่ละคน



รูปที่ 4.17 แสดงการตรวจสอบความเป็นสมาชิก

รูปที่ 4.17 คือมีการอัปเดตทุกวันเพื่อตรวจสอบความเป็นสมาชิกของลูกค้า

ในส่วนของหน้าดังนี้มีการทำงานคือเป็นการตรวจสอบสมาชิกว่าหมดอายุหรือไม่ คือเข้าไปที่ Adjustment Member แล้วก็ทำการเลือกสมาชิกแล้วก็กดตกลงก็จะทำการตรวจสอบข้อมูลการเป็นสมาชิกว่าหมดอายุหรือไม่

## บทที่ 5

### สรุปผลการทดลอง

ปริญญานิพนธ์นี้ได้นำเสนอระบบจัดการแพปลา ซึ่งจากการที่ได้เก็บรวบรวมข้อมูลเพื่อนำข้อมูลที่ได้มาทำการเขียนเป็นโปรแกรมเพื่อเอื้อประโยชน์ แก่ผู้ประกอบการและสามารถนำเทคโนโลยีที่ได้จัดทำขึ้นไปใช้ในองค์กรเพิ่มความสะดวกในการทำงานให้เกิดประสิทธิภาพสูงสุด

ผลที่ได้รับจากการทำปริญญานิพนธ์ คือ ในส่วนของผู้คณะจัดทำเองได้รับความรู้ ในด้านการเขียนโปรแกรมคือได้มีการศึกษา การใช้ ภาษา Visual Basic.Net , SQL Server 2000 และการวิเคราะห์ออกแบบฐานข้อมูล

ในส่วนของตัวโปรแกรมได้มีการออกแบบในส่วนการใช้งานต่างๆ เพื่อรองรับการใช้งานในระบบแพปลา มีส่วนของ เรือ ในส่วนของเรือ จะเก็บข้อมูลคือ ชื่อเรือ ที่อยู่ เจ้าของ วันที่ออกจับปลา วันที่เข้าท่า จำนวนของปลาที่สามารถจับได้แต่ละครั้ง ในส่วนของข้อมูลสัตว์น้ำที่จับได้ จะแยกเป็นประเภท จะเก็บข้อมูลที่รับซื้อจากเรือประมง และราคาที่เขาให้กับลูกค้า ในส่วนของลูกค้าที่มาซื้อ ก็จะมีลูกค้าที่เป็นสมาชิก โดยมีสมาชิกแบบธรรมดา และสมาชิกแบบ VIP